中 華 民 國 專 利 公 報 (19)(12)

(11)公告編號：330366

(44)中華民國87年(1998)04月21日

發 明 全 10 頁

(51)Int.Cl.[6]：H04N9/79

(54)名 稱：具有記錄閉路字幕資料於數位視訊輸入界面之類比視訊編碼器

(21)申 請 案 號：86106026 (22)申請日期：中華民國86年(1997)05月06日

(72)發 明 人：
詹姆士·T·培特森 美國

(71)申 請 人：
三星資訊系統公司 美國

(74)代 理 人：蔡坤財 先生

1

[57] 申請專利範圍：

1. 一種用以解碼一輸入傳輸信號之裝置，包含：

一傳輸解多工器，其接收一輸入傳輸串流並解多工該輸入傳輸串流成為一音訊串流及一包含壓縮視訊及使用者資料之視訊串流；

一視訊解碼器，其接收該視訊串流並由其上產生一解壓縮視訊及閉路字幕資料；

一類比視訊編碼器，其接收該解壓縮視訊及閉路字幕資料，並產生一具編碼閉路字幕資料之類比視訊輸出信號。

2. 如申請專利範圍第1項所述之裝置，其中，該類比視訊編碼器具有一單一界面，用以接收該解壓縮視訊及閉路字幕資料。

3. 如申請專利範圍第2項所述之裝置，其中，該閉路字幕資料係被編碼為一輔助資料。

4. 如申請專利範圍第3項所述之裝置，其

2

中，該輔助資料包含碼資料，該資料指示跟隨著該碼資料之閉路字幕資料。

5. 如申請專利範圍第4項所述之裝置，其中，該閉路字幕資料跟隨碼資料一預定

5. 量之位元組。

6. 如申請專利範圍第4項所述之裝置，其中，該閉路字幕資料跟隨碼資料，由碼資料中之資訊所指示之一數量之位元組。

10. 7. 如申請專利範圍第4項所述之裝置，其中，該類比視訊編碼器包含一資料編碼器，其檢測該資料碼並準備一類比視訊編碼器，以接收路隨該碼資料之閉路字幕資料。

15. 8. 如申請專利範圍第7項所述之裝置，其中，該類比視訊編碼器包含一閉路字幕編碼器及一由該資料解碼器所控制之多工器，以提供該閉路字幕資料至該閉路字幕編碼器。

20. 9. 如申請專利範圍第8項所述之裝置，其

3

中，該類比視訊編碼器包含一色度編碼器，其接收來自該多工器之 Cr 及 Cb 資料並放大或縮小該 Cr 及 Cb 資料用作色度編碼。

10. 如申請專利範圍第 9 項所述之裝置，其中，該類比視訊編碼器包含一亮度處理機，其接收來自該多工器之亮度資料並內插該亮度資料。

11. 如申請專利範圍第 3 項所述之裝置，其中，該解壓縮視訊及該閉路字幕資料符合 CCIR-565 標準。

12. 如申請專利範圍第 2 項所述之裝置，其中，該類比視訊編碼器包含一同步信號產生器，其產生至少一為該視訊解碼器所接收之同步信號，該視訊解碼器反應該該至少一同步信號，而送出該閉路字幕資料至類比視訊編碼器。

13. 如申請專利範圍第 12 項所述之裝置，其中，該閉路字幕資料係於一垂直同步期間被該視訊解碼器所送至該類比視訊編碼器。

14. 如申請專利範圍第 12 項所述之裝置，其中，該閉路字幕資料係於一遮沒期間，在一資料位元組之特定前言之後，被該視訊解碼器所送至該類比視訊編碼器。

15. 如申請專利範圍第 12 項所述之裝置，其中，該類比視訊編碼器包含一資料解碼器，其檢測資料位元組，以決定是否出現閉路字幕資料。

16. 如申請專利範圍第 15 項所述之裝置，其中，該類比視訊編碼器包含一閉路字幕編碼器及一由該同步信號產生器所控制之多工器，用以提供由該資料解碼器所決定予以出現之閉路字幕資料給閉路字幕編碼器。

17. 如申請專利範圍第 16 項所述之裝置，其中，該類比視訊編碼器包含一色度編碼器，其接收來自該多工器之 Cr 及 Cb 資料並放大或縮小該 Cr 及 Cb 資料用

4

作色度編碼。

18. 如申請專利範圍第 17 項所述之裝置，其中，該類比視訊編碼器包含一亮度處理機，其接收來自該多工器之亮度資料並內插該亮度資料。

19. 如申請專利範圍第 18 項所述之裝置，其中，該解壓縮視訊及該閉路字幕資料符合 CCIR-601 標準。

20. 一種類比視訊編碼器，用以轉換一數位視訊串流成為類比視訊信號，該編碼器包含：

一單一界面，用以接收包含視訊資料及閉路字幕資料之數位視訊串流：

至少一視訊資料處理機，用以轉換該視訊資料成為類比視訊信號：

一資料解碼器，用以識別於數位視訊串流中之閉路字幕資料；及

一閉路字幕編碼器，用以接收該閉路字幕資料並將該閉路字幕資料插入類比視訊信號中。

21. 如申請專利範圍第 20 項所述之類比視訊編碼器，其中，該單一界面包含一多工器，該多工器具有輸入用以接收該數位視訊串流，多數可選擇之輸出，以及，一選擇輸入，於該選擇輸入上接收一選擇信號，以選擇該多工器之輸出。

22. 如申請專利範圍第 21 項所述之類比視訊編碼器，其中，至少一視訊處理機包含一色度編碼器，其接收來自多工器之第一輸出之視訊資料中之 Cr 及 Cb 資料，放大或縮小該 Cr 及 Cb 資料，以用以色度編碼，並產生色度包束資訊。

23. 如申請專利範圍第 21 項所述之類比視訊編碼器，更包含另一視訊處理機，其係為一亮度處理機，其接收來自多工器之第二輸出並內插該亮度資料。

24. 如申請專利範圍第 23 項所述之類比視訊編碼器，更包含一數位至類比轉換器，其係連接成以接收色度編碼器，亮度處理機及閉路字幕編碼器之輸出並轉換

5

輸出成為類比信號。

25. 如申請專利範圍第 20 項所述之類比視訊編碼器，其中，該為單一界面所接收之閉路字幕資料係被編碼為輔助資料。

26. 如申請專利範圍第 25 項所述之類比視訊編碼器，其中，該輔助資料包含碼資料，該碼資料指示跟隨該碼資料之閉路字幕資料。

27. 如申請專利範圍第 26 項所述之類比視訊編碼器，其中，該閉路字幕資料跟隨該碼資料一預定量之位元組。

28. 如申請專利範圍第 26 項所述之類比視訊編碼器，其中，該閉路字幕資料跟隨碼資料，由碼資料中之資訊所指示之一數量之位元組。

29. 如申請專利範圍第 20 項所述之類比視訊編碼器，其中，該類比視訊編碼器包含一同步信號產生器，其產生至少一同步信號，以計時於數位視訊串流中，閉路字幕資料之送至類比視訊編碼器。

30. 如申請專利範圍第 29 項所述之類比視訊編碼器，其中，該同步信號係被計時，以造成於一垂直同步期間，閉路字幕資料由該類比視訊編碼器所接收。

31. 如申請專利範圍第 29 項所述之類比視訊編碼器，其中，該同步信號係被計時，以造成於一遮沒期間，在一特定前言之位元組之後，閉路字幕資料由該類比視訊編碼器所接收。

32. 一種將閉路字幕資料插入視訊信號之方法，包含：於一類比視訊編碼器之單一資料界面上，接收數位視訊資料及閉路字幕資料；及

使用類比視訊編碼器，以轉換該數位視訊資料及閉路字幕資料成為具有插入閉路字幕之視訊類比信號。

33. 如申請專利範圍第 32 項所述之方法，其中，該接收之步驟包含接收編碼在輔助資料內之閉路字幕資料。

34. 如申請專利範圍第 33 項所述之方法，

6

其中，該輔助資料包含碼資料，其指示跟隨該碼資料之閉路字幕資料。

35. 如申請專利範圍第 34 項所述之方法，其中，該閉路字幕資料跟隨該碼資料一預定量之位元組。

36. 如申請專利範圍第 35 項所述之方法，其中，該閉路字幕資料跟隨該碼資料，由於碼資料中之資訊所指示之位元組量。

37. 如申請專利範圍第 32 項所述之方法，更包含產生至少一同步信號，以計時於數位視訊資料中之閉路字幕資料送至類比視訊編碼器。

38. 如申請專利範圍第 29 項所述之方法，其中，該同步信號係被計時，以造成於一垂直同步期間，該閉路字幕資料為該類比視訊編碼器所接收。

39. 如申請專利範圍第 29 項所述之方法，其中，該同步信號係被計時，以造成於一遮沒期間，在一特定前言資料位元組後，閉路字幕資料為類比視訊編碼器所接收。

圖示簡單說明：

第一圖是用於一電視之依據先前技藝之用以轉換一 MPEG 傳輸串流成為一類比信號之佈局之方塊圖。

第二圖是一方塊圖，描繪第一圖之先前技藝佈局之元件之細節。

第三圖是依據本發明之一實施例構建之用以轉換一傳輸串流成為類比信號之佈局之方塊圖。

第四圖是依據本發明之一實施例構建之類比視訊編碼器。

第五圖是依據本發明之另一實施例構建之類比視訊編碼器之方塊圖。

第六圖是一依據本發明之一實施例構建之一資料解碼器之方塊圖，該解碼器係用於第四圖之類比視訊編碼器中。

第七圖是一依據本發明之一實施例構建之一資料解碼器之方塊圖，該解碼

7

器係用於第五圖之類比視訊編碼器中。

第八圖是一依據本發明之一實施例所構建並被使用於第三圖中之佈局中之資料解碼器之方塊圖，其係被以一於第四及六圖中所描繪之類比視訊編碼器加以構建。

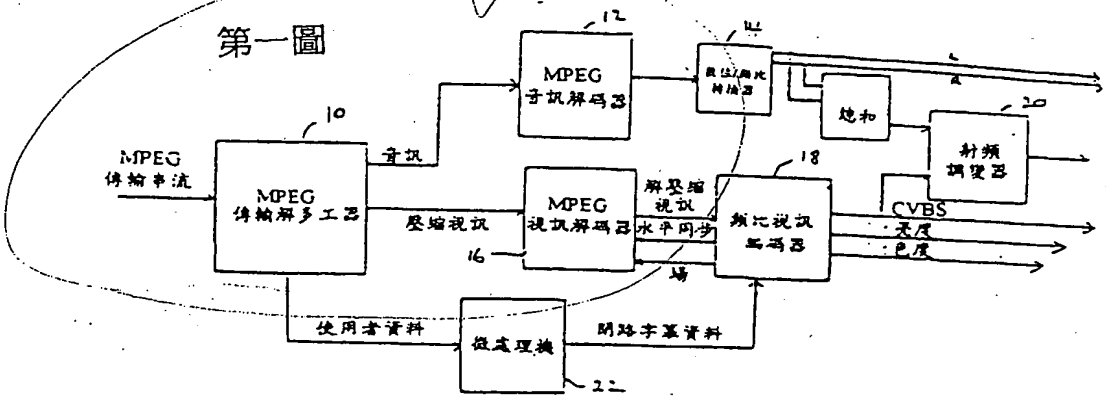
第九圖是一依據本發明之一實施例所構建並被使用於第八圖中之視訊解碼器之閉路字幕資料內佇器之方塊圖；

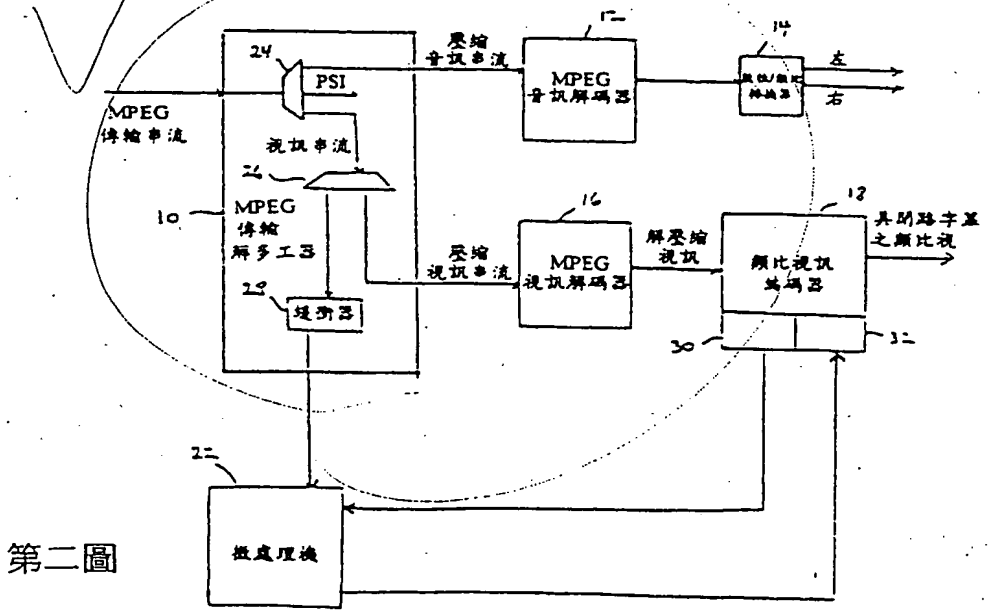
8

第十圖是一依據本發明之一實施例所構建並被使用於第三圖中之具有閉路字幕內佇器之一視訊解碼器之方塊圖，其係被構建以於第五及七圖中所描述之類比視訊編碼器。

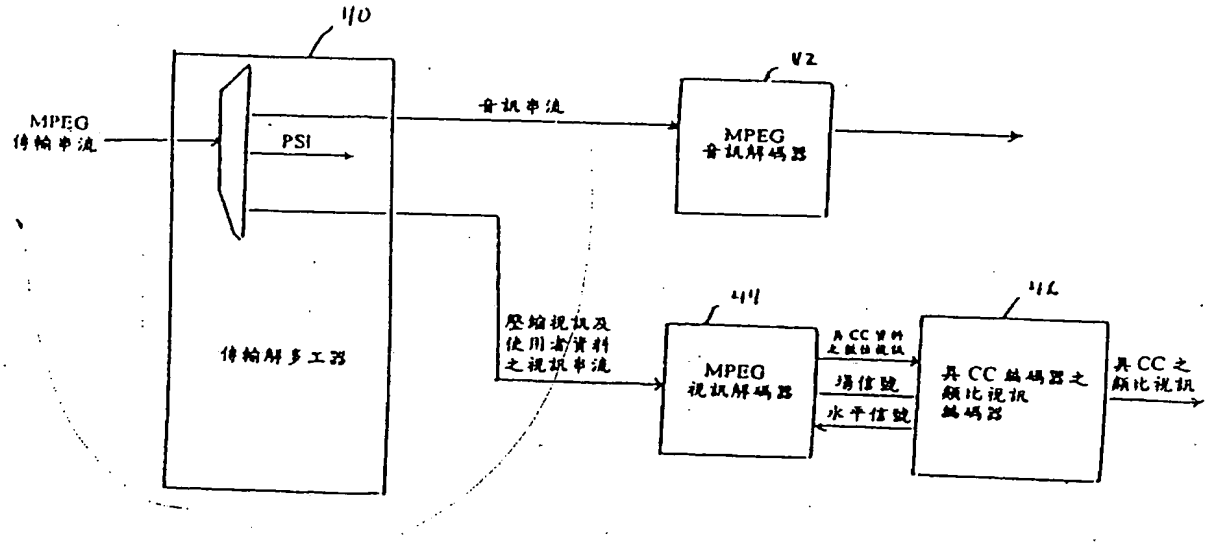
第十一圖是一依據本發明之一實施例所構建並被使用於第十圖之視訊解碼器中之閉路字幕資料內佇器之方塊圖。

第一圖

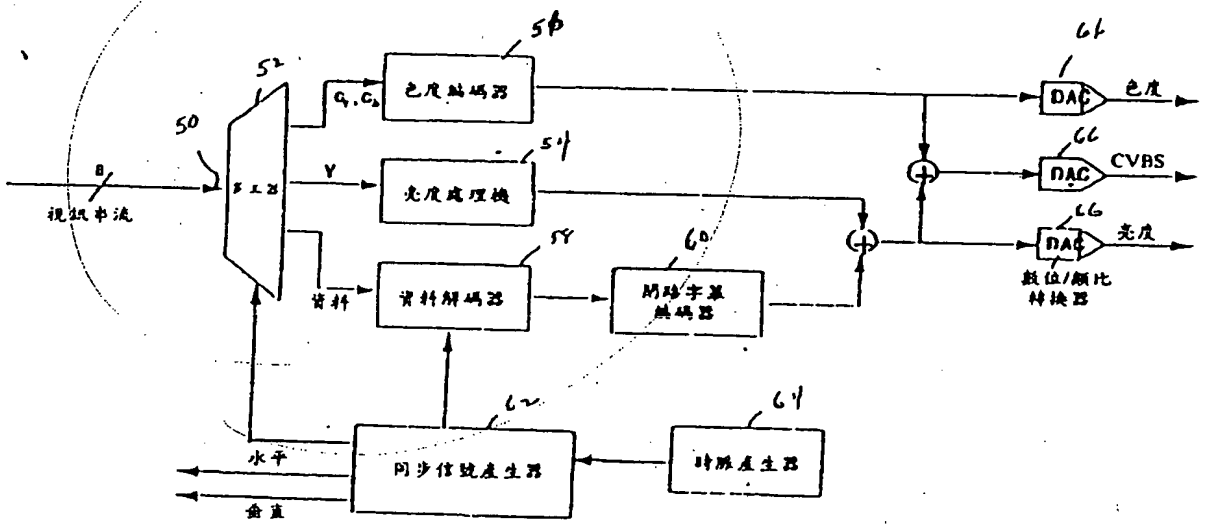




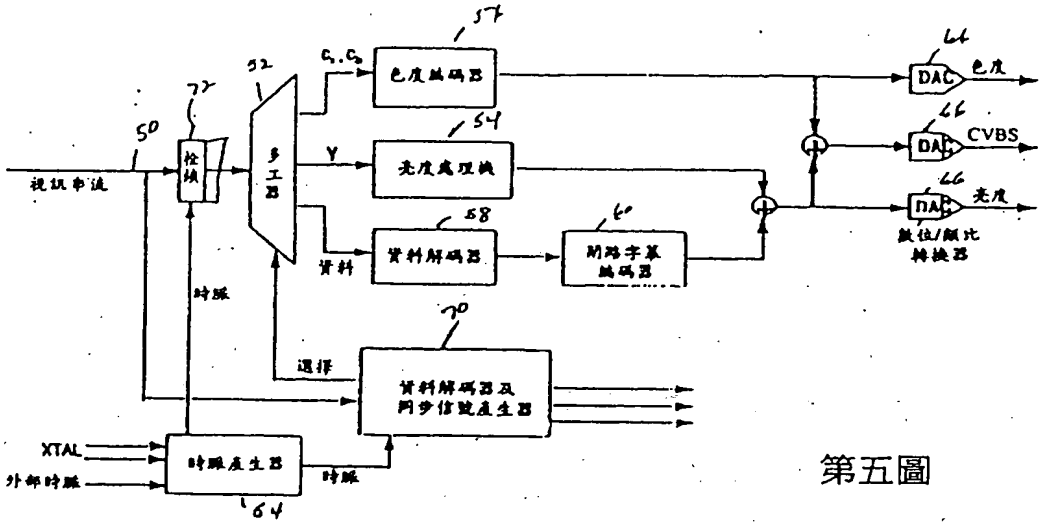
第二圖


MPEG
傳輸串流
PSI
傳輸解多工器
音訊串流
MPEG
音訊解碼器
壓縮視訊及
使用者資料
之視訊串流
MPEG
視訊解碼器
場信號
水平信號
具 CC 編碼器之
類比視訊
編碼器
具 CC 之
類比視訊

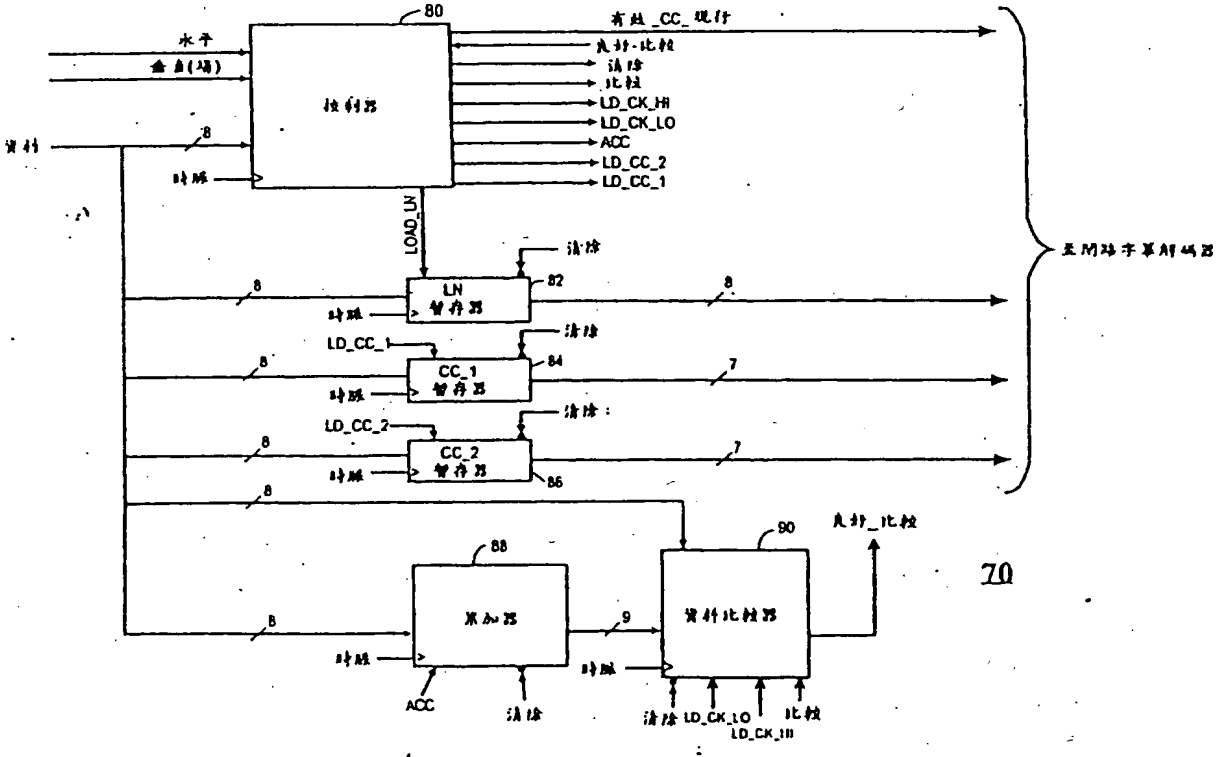
第三圖


8
視訊串流
多工器
Y
色度編碼器
亮度處理機
資料
資料解碼器
閉路字幕
編碼器
同步信號產生器
時脈產生器
水平
垂直
DAC
色度
CVBS
亮度
數位/類比
轉換器

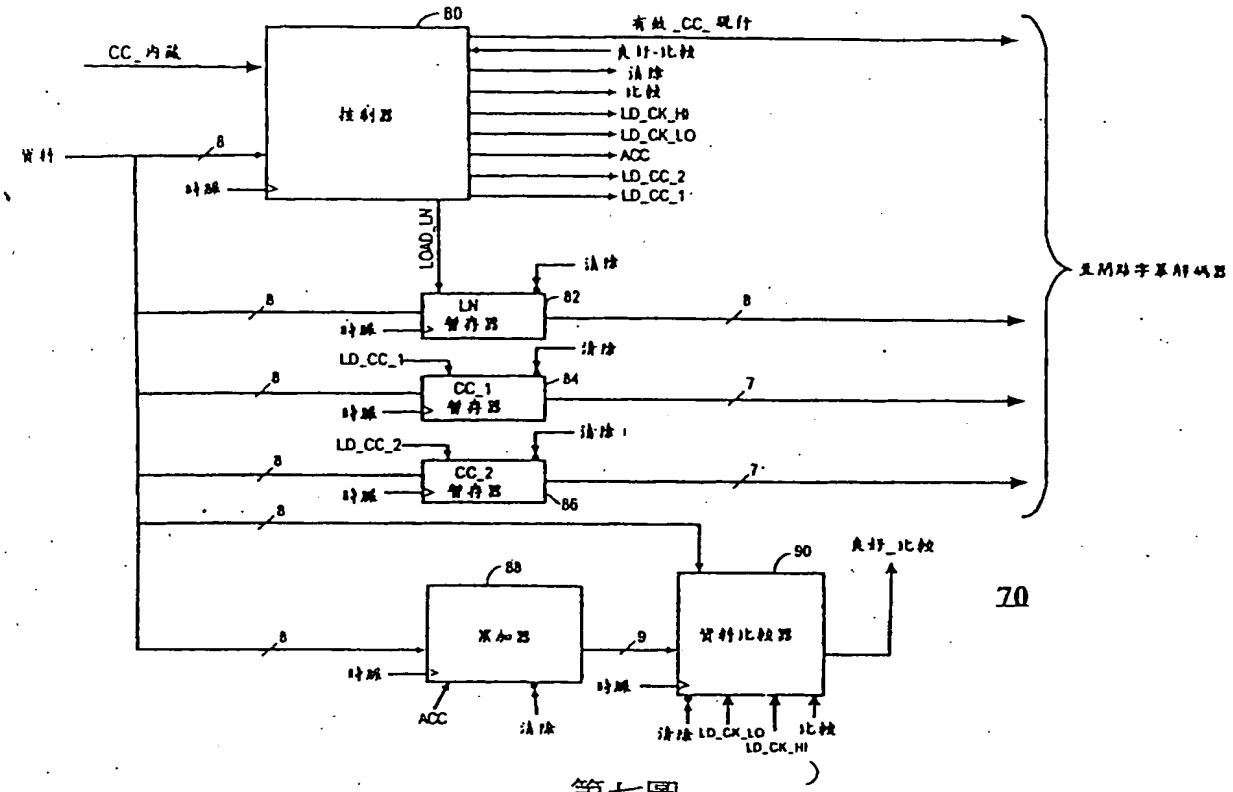
第四圖


視訊串流
檢錯
多工器
色度處理機
亮度處理機
資料解碼器
閉路字幕解碼器
資料解碼器及同步信號產生器
時脈產生器
XTAL
外部時脈
時脈
選擇
資料
DAC
色度
CVBS
亮度
數位/類比轉換器

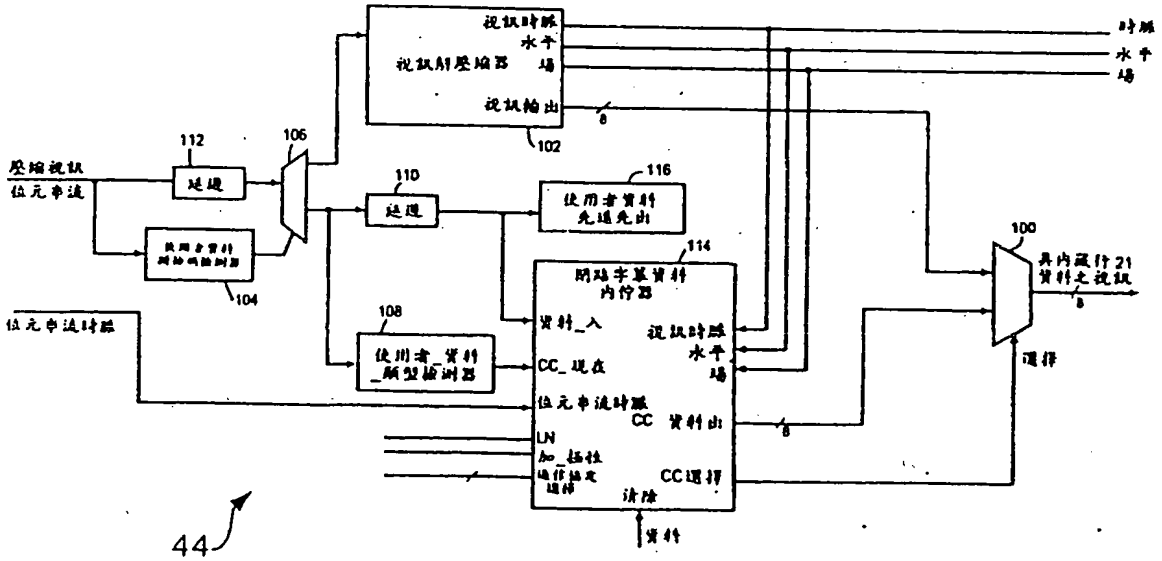
第五圖


水平
垂直(場)
資料
時脈
LOAD_LN
LD_CK_HI
LD_CK_LO
ACC
LD_CC_2
LD_CC_1
清除
LN 暫存器
CC_1 暫存器
CC_2 暫存器
累加器
資料比較器
比較
70

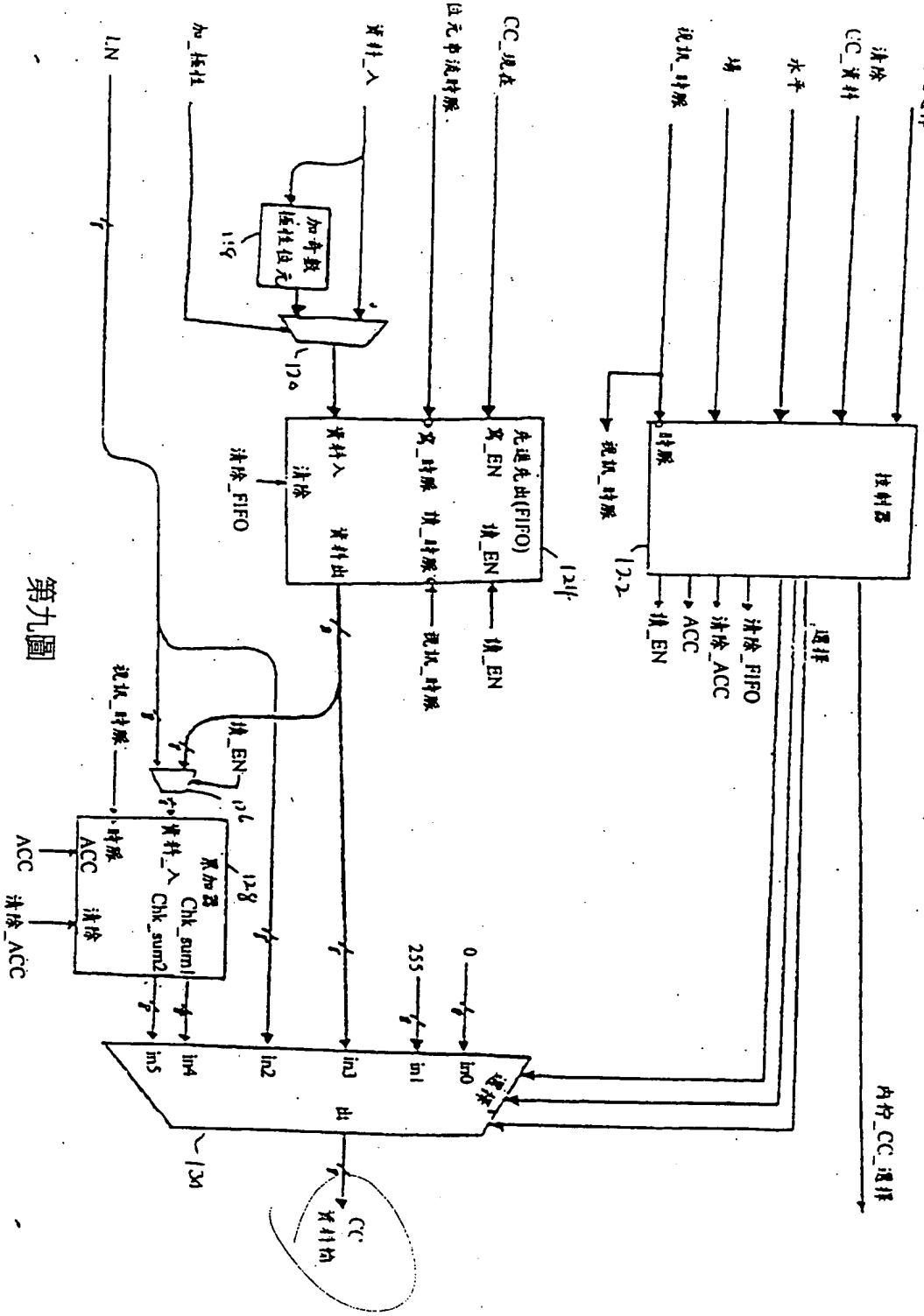
第六圖


80
CC_內設
資料
8
時脈
控制器
有效_CC_現行
真時-比較
清除
比較
LD_CK_HI
LD_CK_LO
ACC
LD_CC_2
LD_CC_1
LOAD_LN
LN 暫存器
82
CC_1 暫存器
84
CC_2 暫存器
86
累加器
88
資料比較器
90
真時_比較
ACC
清除
LD_CK_LO
LD_CK_HI
比較
70

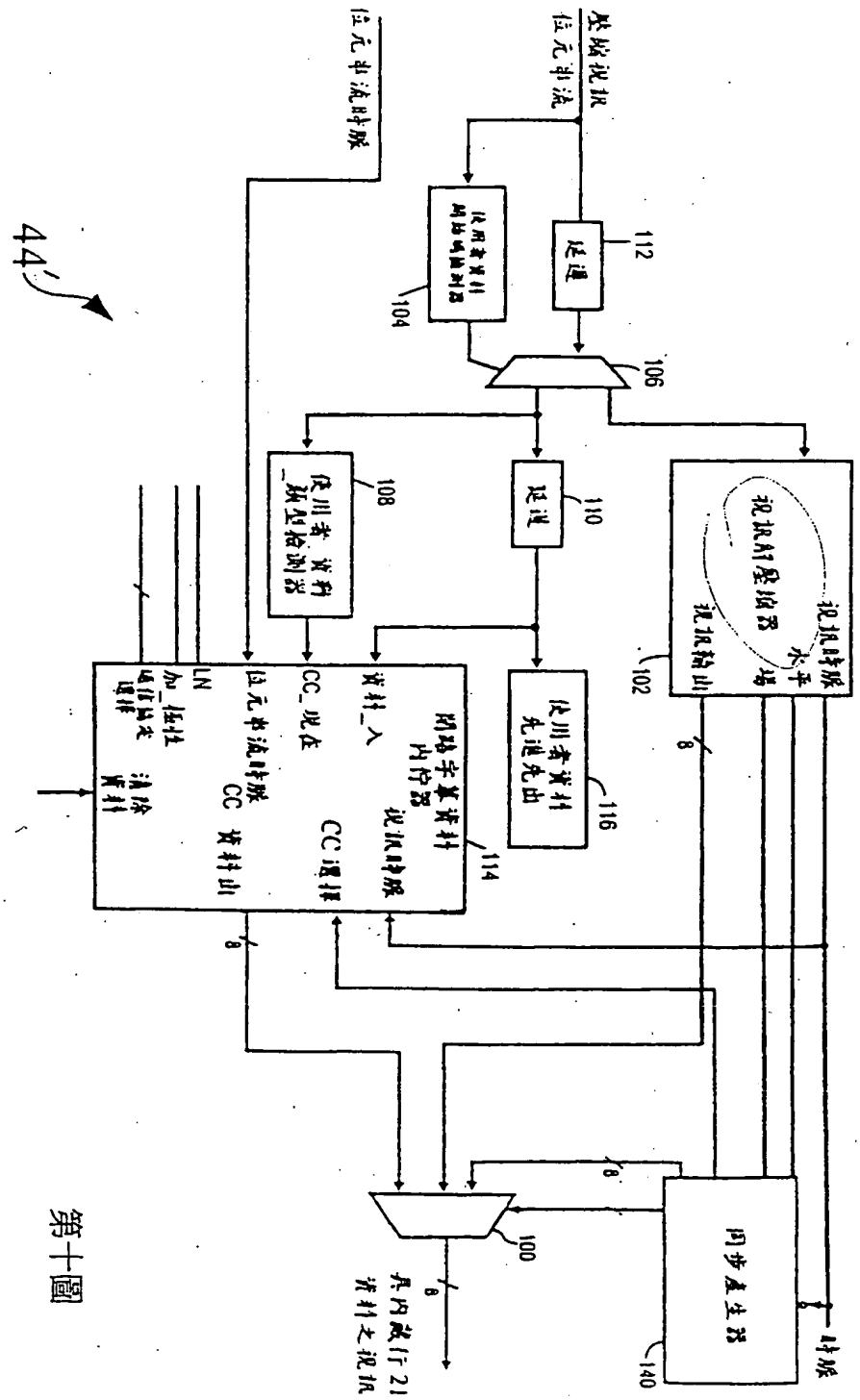
第七圖


視訊時脈
水平
場
視訊解壓縮器
視訊輸出
102
時脈
水平
場
112
106
壓縮視訊
位元串流
延遲
110
延遲
116
使用者資料
先進先出
100
具內設行21
資料之視訊
選擇
104
114
開碼字幕資料
內容器
108
使用者_資料
_類型偵測器
資料_入
視訊時脈
水平
場
CC_現在
位元串流時脈
CC 資料出
LN
加_掃描
CC選擇
清除
位元串流時脈
資料
44

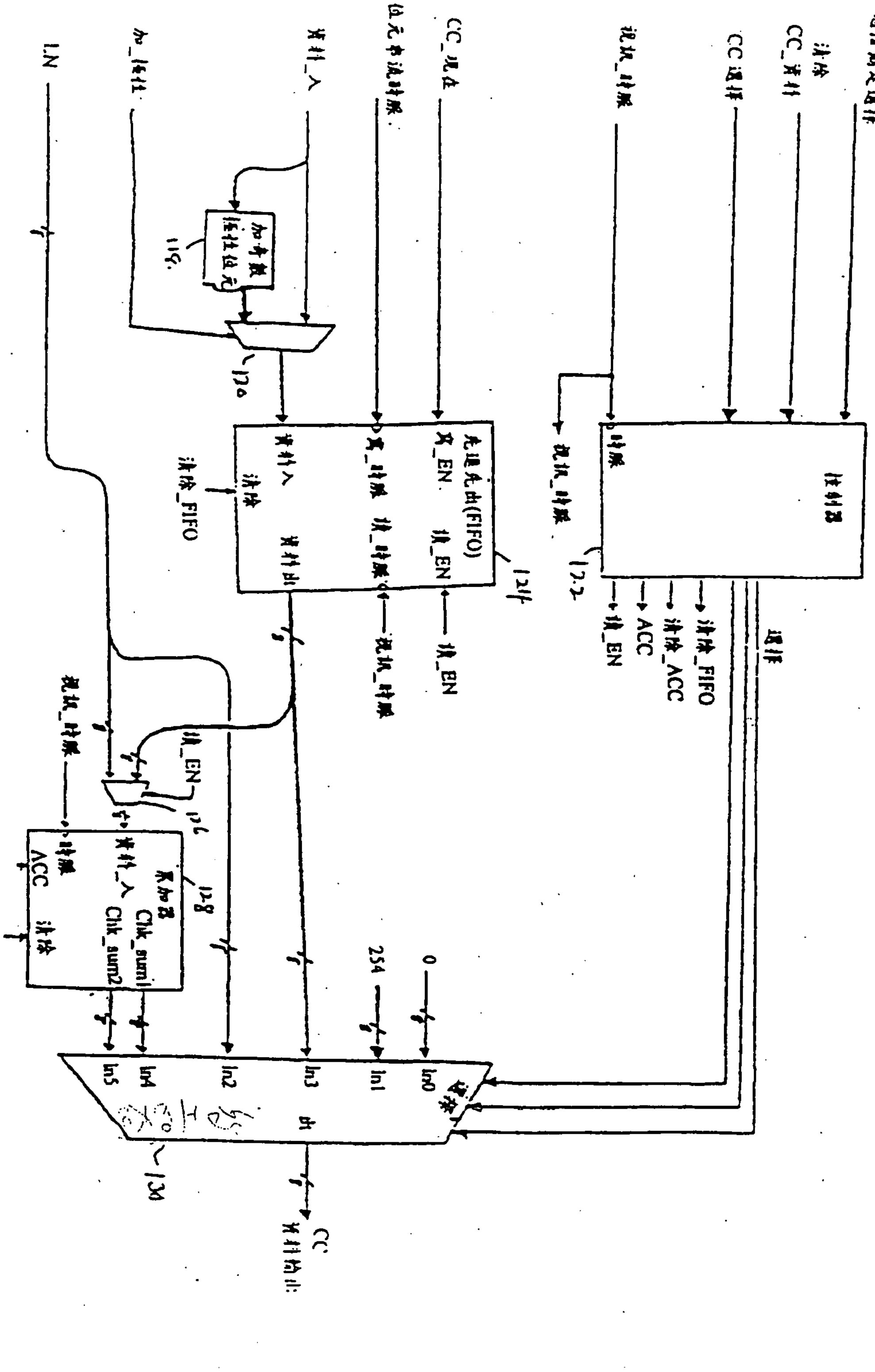
第八圖


通信協定選擇
清除
CC_資料
水平
場
視訊_時脈
控制器
時脈
選擇
清除_FIFO
清除_ACC
ACC
填_EN
視訊_時脈
內存_CC_選擇
CC_現在
位元串流時脈
資料_入
加_填位
加奇數
填位位元
1:9
130
資料入
清除
寫_時脈
先進先出(FIFO)
寫_EN
填_EN
資料出
清除_FIFO
現訊_時脈
124
填_EN
現訊_時脈
填_EN
126
資料_入
Chk_sum1
Chk_sum2
128
時脈
ACC
清除
ACC
清除_ACC
1:N
in5
in4
in2
in3
in1
in0
255
0
選擇
出
134
CC
資料流

第九圖


44
位元串流
位元串流時脈
延遲
112
104
106
延遲
110
108
102
116
114
100
140
8
LN
CC_現在
CC 資料出
CC 選擇
資料_入
時脈

第十圖


先進先出(FIFO)
控制器
清除_FIFO
清除_ACC
ACC
寫_EN
讀_EN
累加器
Clk_sum1
Clk_sum2
In0
In1
In2
In3
In4
In5
254
0
CC

第十一圖

2810-005

ANALOG VIDEO ENCODER WITH METERED CLOSED CAPTION DATA ON DIGITAL VIDEO INPUT INTERFACE

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention relates to the field of video data processing, and more particularly, to an apparatus and a method for performing analog video encoding of digital video data with closed caption data.

DESCRIPTION OF RELATED ART

Federal Regulations require that televisions of a specified size have closed captioning capability. Broadcasters encode closed captioning data in portions of the video that are broadcast. Televisions decode the closed captioning data and display it on the screen with the video information. Closed captioning data assists the hearing impaired and people who do not have English as their native language to enjoy television. Closed caption encoding has been performed in the analog domain since television broadcasts have also been in the analog domain.

Relatively recent advances in technology have provided a digital means of transmitting video and audio data using image compression. As an example, the MPEG standard (Motion Pictures Experts Group) is a multimedia video playback standard that allows digital video to be compressed using a combination of image compression and

a sophisticated form of differencing. A video sequence is encoded by recording differences between frames rather than entire images of each frame. The MPEG-2 standard supports high-quality digital video.

The MPEG standard was developed independently of television and did not originally consider closed captioning. As the desirability of delivering information in digital form to televisions has increased, features have been added to the MPEG standard in order to make it compatible with the requirements of television. These include the feature of providing closed captioning in the display of the data provided in the MPEG data stream. The recommended procedure for encoding closed captioning in the MPEG video stream is to provide the closed captioning data within the "user data" in the MPEG data stream. A block diagram of a prior art system for converting the input MPEG stream to an analog signal usable by a television is shown in the block diagram of Figure 1.

The MPEG transport stream is provided to an MPEG transport demultiplexer (demux) 10. The MPEG transport stream is separated into an audio signal, a compressed video signal, and a user data signal. The audio signal is decoded by an MPEG audio decoder 12 and converted by a digital to analog converter 14 into a stereo analog signal. The compressed video signal is received by a video decoder 16 that decompresses the video signal according to a specified standard, such as CCIR-601 or CCIR-656. The decompressed video signal is converted to an analog signal by an analog video encoder 18, which provides a horizontal sync signal and a field signal to the video decoder 16. The analog signals generated by the analog video encoder 18 are the CVBS signal, the Y signal, and the C signal. The audio and the video signals are provided to a television set (not depicted in Figure 1) through an RF modulator 20.

The user data is separated from the compressed video data and provided to the analog video encoder via a separate interface, as depicted in Figure 1. The user data is read by a microprocessor 22 which first checks the analog video encoder 18 to determine whether the original analog video encoder 18 is able to accept the closed caption data at that time.

Figure 2 is a more detailed depiction of portions of the prior art arrangement of Figure 1. The transport demultiplexer 10 is depicted with multiplexers 24 and 26. The transport demultiplexer 10 generates an interrupt when it receives the user data. The microprocessor receives that interrupt, retrieves the user data from a buffer 28 and temporarily stores it. The microprocessor 22 monitors the status of the analog video encoder 18 through a status register 30. The contents of the status register 30 provide an indication to the microprocessor 22 whether the analog video encoder 18 is able to receive closed captioned data. When the status register 30 indicates that the analog video encoder 18 is ready to receive closed captioned data, the microprocessor 22 forwards the user data originally stored in buffer 28 to the closed captioned data register 32 and the analog video encoder 18. Closed captioned data is normally sent when the vertical retrace signal is sent out by the analog video encoder 18.

The use of a microprocessor in the prior art, as depicted in Figures 1 and 2, has the disadvantage of requiring extra hardware (the microprocessor) and an extra layer of software to run the microprocessor in order to perform the function of inserting closed captioned data in the video stream. This extra layer of hardware and software to insert the closed captioned data within an analog video stream complicates the flow of data and adds expense to the device.

SUMMARY OF THE INVENTION

There is a need for an analog video decoder and method for inserting closed captioned data contained within a digital video stream, such as an MPEG transport stream, into an analog output stream without the use of a microprocessor or its associated software.

This and other needs are met by the present invention which provides an arrangement for decoding an input transport signal, comprising a transport demultiplexer that receives an input transport stream and demultiplexes the input transport stream into an audio stream and a video stream containing compressed video and user data. The arrangement also includes a video decoder that receives the video stream and generates decompressed video and closed captioned data from the video stream. An analog video encoder is provided that receives the decompressed video and closed captioned data and generates an analog video output signal with encoded closed captioned data. The analog video encoder has a signal interface for receiving a decompressed video and closed captioned data.

The arrangement of the present invention avoids the use of a microprocessor to poll the analog video encoder, as well as the software needed to operate the microprocessor, by providing an analog video encoder that has a single interface that receives both the decompressed video and the closed captioned data.

In another aspect of the present invention, an analog video encoder for converting a digital video stream to analog video signals is provided. The analog video encoder comprises a single interface for receiving a digital video stream that includes video data and closed captioned data. At least one video data processor is provided for converting the video data into analog video signals. A data decoder identifies the closed captioned data in the digital video stream, and a closed

captioning encoder receives the closed captioned data and inserts the enclosed captioned data into the analog video signals.

In a still further aspect of the present invention, a method of inserting closed captioned data in a video signal is provided. The method comprises the steps of receiving digital video data and closed captioned data at a single data interface of an analog video encoder. The digital video data and closed captioned data are converted into video analog signals with inserted closed captioning using the analog video encoder.

The foregoing and other features, aspects and advantages of the present invention will become more apparent from the following detailed description of the present invention when taken in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram of an arrangement according to the prior art for converting an MPEG transport stream into analog signals for use by a television.

Figure 2 is a block diagram depicting in more detail elements of the prior art arrangement of Figure 1.

Figure 3 is a block diagram of an arrangement for converting a transport stream into analog signals constructed in accordance with an embodiment of the present invention.

Figure 4 is an analog video encoder constructed in accordance with an embodiment of the present invention.

Figure 5 is a block diagram of an analog video encoder constructed in accordance with another embodiment of the present invention.

Figure 6 is a block diagram of a data decoder constructed in accordance with an embodiment of the

present invention for use in the analog video encoder of Figure 4.

Figure 7 is a block diagram of a data decoder constructed in accordance with an embodiment of the present invention for use in the analog video encoder of Figure 5.

Figure 8 is a block diagram of a video decoder with closed caption embedder constructed in accordance with an embodiment of the present invention and used in the arrangement of Figure 3, when configured with an analog video encoder as depicted in Figures 4 and 6.

Figure 9 is a block diagram of a closed caption data embedder constructed in accordance with an embodiment of the present invention and used in the video decoder of Figure 8.

Figure 10 is a block diagram of a video decoder with closed caption embedder constructed in accordance with an embodiment of the present invention and used in the arrangement of Figure 3, when configured with an analog video encoder as depicted in Figures 5 and 7.

Figure 11 is a block diagram of a closed caption data embedder constructed in accordance with an embodiment of the present invention and used in the video decoder of Figure 10.

DETAILED DESCRIPTION OF THE ILLUSTRATIVE EMBODIMENTS

Figure 3 is a block diagram of a conversion arrangement for converting a transport stream, such as an MPEG transport stream, into analog signals suitable for use by a conventional television set.

The arrangement includes a transport demultiplexer 40 which demultiplexes the transport stream, such as an MPEG transport stream, into an audio stream and a video stream. The MPEG transport stream also includes a PSI stream.

The demultiplexed audio stream is provided to an MPEG audio decoder 42, which converts the digital audio stream into an analog audio signal. The video stream, which comprises compressed video and user data, is sent to an MPEG video decoder 44. The video stream is decoded by the MPEG video decoder 44 into a decompressed digital video stream with closed captioned data. Exemplary embodiments of a video decoder usable in the arrangement of Figure 3 is described in more detail later and are depicted in Figures 8-11. The decompressed video and closed captioned data is received at a single input port interface at an analog video encoder 46 that includes a closed caption encoder. The analog video encoder 46 converts the digital video signal and inserts the closed captioned data sent with the digital video signal and received at the single interface of the analog video encoder 46 into analog signals usable by a conventional television receiver.

The arrangement of Figure 3 avoids the use of a microprocessor which polls a status register in the analog video encoder 46 and supplies closed caption data from the video decoder 44 to a separate interface of the analog video encoder 46. The elimination of a microprocessor and a separate interface for the closed captioned data at the analog video encoder reduces the cost of the arrangement (due to the decrease in hardware). The complexity of the arrangement is also reduced since the software to operate the microprocessor to poll the analog video encoder to properly time the sending of the closed captioned data to the analog video encoder is eliminated.

An embodiment of the analog video encoder 46 depicted in Figure 3 is provided in the block diagram of Figure 4. The analog video encoder of Figure 4 is suitable for converting a decompressed video signal that conforms to the CCIR-601 standard. Currently the CCIR-

601 mode is the most popular in MPEG systems because the analog video encoder is used as the display synchronization master. This means that the analog video encoder generates the horizontal and field signals for the MPEG video decoder 44 of Figure 3.

The video data stream received at the single video input interface 50 of the analog video encoder 46 is an 8-bit video stream, for example. The video data stream is in CCIR-601 mode with closed caption data added. The CCIR-601 video data is multiplexed luma (Y) and chroma (Cr, Cb). It is in the sampling format referred to as 4:2:2, which means that the active video signal of each line of video appears in the following order: Cb, Y, Cr, Y, Cb, Y, Cr, Y,... During the active video time, the multiplexer 52 of the analog video encoder 46 sends the multiplexed luma samples through a luma processor 54. Also during this active video time, the multiplexer 52 provides the chroma (Cr, Cb) to a chroma encoder 56. The multiplexer 52 sends the video data stream to a data decoder 58 during the non-active video time.

The luma processor 54 receives the luma signal arriving at 13.5 MHz rate and interpolates the signal to 27 MHz. This eases design requirements of analog anti-imaging filters on digital to analog convertor outputs. In addition, any luma filtering such as bandwidth reduction or chroma notch filtering, is performed at the luma processor 54.

The chroma encoder 56 receives the Cr and Cb data and scales this data properly for chroma encoding. The chroma encoder 56 also generates and asserts color bursts into the output signal. The chroma bandwidth is also reducible using the chroma encoder 56.

The data decoder 58 examines the video data stream sent to it by the multiplexer 52 for a sequence of bytes that indicates the presence of closed captioned data. Depending on the protocol established between the MPEG

video decoder 44 and the analog video encoder 46, the sequence of bytes occurs at a certain time during a non-active video time.

As an example, assume that the sequence is 0, 255, LN, CC_1, CC_2, Chk_sum1, and Chk_sum2. In this sequence, 0, 255 is the start sequence. LN is the line number where the closed captioned data is to be encoded. CC_1 is the first byte of closed captioned data. CC_2 is the second byte of closed captioned data. Chk_sum1 is the seven most significant bits of the check sum and Chk_sum2 is the seven least significant bits of the check sum. The upper bit of Chk_sum1 and Chk_sum2 are set to 0 to prevent false start sequence emulation. The start sequence is unique since, according to the EIA608 recommendation, all of the closed captioned data types are ASCII. They are also limited to a range from 16 to 127.

Figure 6 is a block diagram of a data decoder constructed in accordance with an embodiment of the present invention that may be used in the embodiment of the analog video encoder of Figure 4. A controller 80 receives the data stream (CCIR-601 data), the horizontal timing reference signal and the vertical (field signal) timing reference. When the controller 80 recognizes a transition on the field signal, it clears an LN register 82, a CC_1 register 84, a CC_2 register 86, an accumulator 88, and a data comparator 90 by setting the clear signal low for one clock cycle. The controller 80 also sets the following output signals low: Valid_CC_present, COMP, LD_CK_HI, LD_CK_LO, ACC, LD_CC_1, and LD_CC_2.

At the next falling edge of the horizontal signal after the transition of the field signal the controller 50 starts looking for the start sequence. After the controller 80 receives the start sequence of 0, 255, it enables the LN register 82 by setting LOAD_LN high.

On the next clock, the controller 80 enables loading of the CC_1 register 54 by setting LD_CC_1 high. The controller 80 also enables the accumulator 88 by setting ACC high. The LN register 82 is disabled by the controller 80 setting LOAD_LN low.

On the next clock, the controller 80 enables loading of the CC_2 register 86 by setting LD_CC_2 high and disables CC_1 register loading by setting LD_CC_1 low. On the next clock, the controller 80 disables the accumulator 88 by setting ACC low. The CC_2 register loading is disabled by the controller 80 setting LD_CC_2 low. The controller 80 also instructs the data comparator 90 to load the first byte of the check sum (Chk_sum1) by setting LD_CK_LO high. On the next clock, the controller 80 instructs the data comparator 90 to load the second byte of the check sum (Chk_sum2) by setting LD_CK_HI high and setting LD_CK_LO low.

On the next clock, the controller 80 instructs the data comparator 90 to compare the check sum (formed with Chk_sum1 and Chk_sum2) with the nine-bit value from the accumulator 88. This is accomplished by the controller 80 setting LD_CK_HI signal low and setting the COMP signal high. If the values compare "OK", the data comparator 90 sets the GOOD_COMPARE signal high. If the values do not compare "OK", then the data comparator 90 maintains the GOOD_COMPARE signal low.

In response to the GOOD_COMPARE signal being high, the controller 80 on the next clock sets the Valid_CC_present signal high. If the GOOD_COMPARE signal is low, then the controller 80 leaves the Valid_CC_present signal low and clears the LN register 82, the CC_1 register 84, the CC_2 register 86, the accumulator 88 and the data comparator 90 by setting the CLEAR signal low for one clock cycle. The controller 80 then returns to examining for a field signal transition to begin the process again.

The protocol may be a three step process. For example, the analog video encoder 46 may make a transition on the field signal. When the analog video encoder 46 generates the next falling edge of the horizontal signal, the analog video encoder 46 expects to receive the sequence of bytes described above at its digital video input interface 50. If the above sequence of bytes is not seen, then the data decoder 58 does not encode data in the next line 21.

Referring to Figure 4, a closed captioned encoder 60 is coupled to the output of the data decoder 58. When enabled, the closed captioned encoder 60 automatically generates seven cycles of clock running (32 X line rate) start bit insertion (001) and finally insertion of the two data bytes per line. Data bits are encoded so that logic low at the video outputs corresponds to zero IRE and logic high corresponds to 50 IRE. This conforms to Figure 1, section 2.2 of standard EIA-608. The EIA-608 is the recommended practice for line 21 data service.

The analog video encoder 46 includes a synchronization signal generator 62 that receives a 27 MHz clock signal from clock generator 64 and generates all of the horizontal and vertical timing references. Included in the horizontal references are the "start of active video," "end of active video," and "burst gate flag". The synchronization signal generator 62 provides the external horizontal and field signals to the MPEG video decoder 44. The select signals for the multiplexer 52 are also provided by the synchronization signal generator 62.

The clock generator 64 generates a clock signal that is 32 times the line rate clock for the closed captioned encoder. This clock signal is locked to the horizontal timing reference.

The outputs of the chroma encoder 56, luma processor 54, and the closed captioned encoder 60 are provided as

inputs to digital to analog converters (DAC's) 66. These digital to analog converters convert the digital signals to analog chroma, CVBS, and luma signals suitable for use by a conventional television apparatus.

In operation of the analog video encoder 46 of Figure 4, the synchronization signal generator 62 will send a synchronization signal, either the horizontal or field signal, to the video decoder 44 (depicted in Figure 3). In response, the video decoder 44 will order two bytes of closed captioned data within a predetermined time (i.e., a certain number of data bytes) after the synchronization signal has been sent by the synchronization signal generator 62. The synchronization signal generator 62 will, at the appropriate time after sending its synchronization signal to the video decoder 44, control the multiplexer 52 to provide the video data to the data decoder 58. The synchronization signal generator 62 will also control the data decoder 58 at this time. The data decoder 58 will search for the sequence of bytes that indicates closed captioned data is present. If the data decoder 58 determines that the closed captioned data is present, it provides this data to the closed captioning encoder 60, which automatically generates the appropriate cycles of clock running, start bit insertion and insertion of the two data bytes per line. The signal output by the closed captioned encoder 60 is combined with the signal from the luma processor 54, and the chroma encoder 56.

According to certain embodiments of the present invention, the closed captioned data is sent to the analog video encoder following a specific preamble of data bytes during a blanking interval. In other embodiments, the closed captioned data is sent to the analog video encoder by the video decoder 44 during a vertical synchronization period.

Figure 5 is a block diagram of another embodiment of the analog video encoder 46, this embodiment being suitable for converting data that is in CCIR-656 mode. A CCIR-656 data is a super set of CCIR-601. The data includes horizontal timing information, vertical timing information and ancillary data. Since the video signal itself contains the horizontal timing information and vertical timing information (as well as the ancillary data), the analog video encoder 46 does not act as the display synchronization master. Instead, the data decoder and synchronization signal generator 70 in the embodiment of Figure 5 receives the video stream directly at the single video interface 50. The synchronization signal generator 70 also has a data decoder which examines the video stream to determine whether the ancillary data contains closed captioning data. If so, the data decoder and synchronization signal generator 70 will cause the multiplexer 52 to send the ancillary data to the data decoder 58.

The chroma encoder 56, luma processor 54 and closed captioning encoder 60 operate in the same manner as in the embodiment of Figure 4 and their description of operation will not be repeated.

A latch 72 is provided at the single input interface 50 and before the multiplexer 52 to latch the video stream as it is being input into the analog video encoder 46. This provides the data decoder and synchronization signal generator 70 with time to examine the video stream to determine the selection of the multiplexer and correctly route the data between the chroma encoder 56, luma processor 54 and data decoder 58.

In operation, the video stream contains the ancillary data that has closed captioned data within it. The ancillary data may include code data which indicates that closed captioned data follows the code data within the ancillary data. This is detected by the data decoder

and synchronization signal generator 70 while the video stream is latched in latch 72. When the data decoder and synchronization signal generator 70 identifies a 0, 255, and timing information byte sequence in the data stream to the analog video encoder 46, the generator 70 becomes aware that it is receiving ancillary data. The generator then waits for a byte sequence that indicates that the ancillary data has closed caption data embedded. Once the generator 70 identifies that byte sequence, it informs the data decoding module 58 that closed captioned data is embedded by setting a CC_Embedded signal high. Data decoder and synchronization signal generator 70 causes the multiplexer 52 to route the video stream to the data decoder 58. In certain embodiments, the closed captioned data follows the code data by a predetermined amount of bytes. In other embodiments, the closed captioned data follows the code data by an amount of bytes (or lines) indicated by information in the code data. This allows the timing of the receipt of the closed captioned data to be variable.

The data decoder of Figure 7 is similar to that of Figure 6, and therefore its components have the same reference numerals as used in Figure 6. However, the controller 50 in the embodiment of the data decoder of Figure 7 is responsive to the CC_Embedded signal asserted by the data decoder and synchronization signal generator 70, instead of the field signal transition. The detailed operation of the data decoder of Figure 7 is otherwise the same as that of Figure 6.

An exemplary embodiment of a video decoder 44 for use with the analog video encoder of Figure 4 is depicted in Figure 8. Accordingly, the decompressed digital video that is produced at the output of this embodiment of the video decoder is in the CCIR-601 mode. In this mode, the analog video encoder 46 is used as the display

synchronization master, and the CCIR-601 mode bitstream is just digital video.

As an overview of its functionality, the video decoder 44 receives closed caption data (or other data for encoding in line 21 of the vertical blanking interval of analog video) embedded in the user data portion of the compressed video bitstream. If the video decoder 44 determines that the user data is closed caption data (or other data for line 21 encoding), then the video decoder 44 will send out the data at a rate of two bytes per field, on the same data interface used for the decompressed digital video output. For the embodiment of Figure 10, in which the output digital video is CCIR-656 data, then the closed caption data may be encoded as ancillary data. If the output digital video is CCIR-601, then the closed caption data may be presented at a certain time (such as during the vertical sync) or after a certain preamble of bytes during a blanking interval.

In the CCIR-601 mode, the video decoder 44 is not the display synchronization master. This means that the display horizontal and field synchronization signals are generated externally for the video decoder 44. As described earlier, in the CCIR-601 mode the video data is multiplexed luma (Y) and chroma (Cr, Cb). It is in the sampling format referred to as 4:2:2, which means that the active video signal of each line of video appears in the following order: Cb, Y, Cr, Y, Cb, Y, Cr, Y,... During the active video time, the second multiplexer 100 of the video decoder 44 of Figure 8 sends the Cb, Y and Cr data from a video decompressor 102 to the outputs of the video decoder 44. During the non-active video time, the second multiplexer 100 is able to send the byte sequence that contains closed caption data to the video output of the video decoder 44.

A user data start code detector 104 scans the compressed video bitstream for user data start codes.

User data may be in the sequence layer, the group of picture (GOP) layer, or the picture layer, as described in Motion Picture Experts Group Video, ISO/IEC 13818-2:1995, Information Technology--Generic coding of moving pictures and associated audio information: Video. When the user data start code detector 104 sees user data start sequences, the detector 104 controls a first multiplexer 106 to send the bitstream to a user data type detector 108 and a second delay 110. A first delay 112 compensate for the processing delay of the user data start code detector 104.

The user data type detector 108 examines the user data type field in the user data bitstream to determine if it is closed caption data (or data for line 21 encoding). The user data type field will be 08 (in hexadecimal format) if the user data is closed caption data. The field will be 09 (in hexadecimal) if it is data for line 21 encoding. If the user data is closed caption or data for line 21 encoding, then the user data type detector 108 informs a closed caption data embedder 114 that the data it is receiving should be processed by setting CC_present signal high. The second delay 110 compensates for the processing delay of the user data type detector 108.

The video decompressor 102 receives the compressed video bitstream and produces uncompressed digital video in a conventional manner. The digital video output of the video decompressor 102 has CCIR-601 levels, in which the luma values range from 16 to 235, the Cr and Cb value range from 16 to 240.

A user data FIFO (first-in, first-out buffer) 116 holds all of the user data that is received in the compressed video bitstream. The user data may contain other data that is not closed-caption data that the rest of the system may need.

The closed caption data embedder 114 receives the closed caption user data and sends it out on the CC_data_out port. An embodiment of the data embedder 114 used for the CCIR-601 mode (where the video decoder 44 is slaved to the external video synchronization signals) is depicted in Figure 9.

When the CC_present signal is high, the embedder 114 knows that the bitstream data at the Data_in port is closed caption data or data for line 21 encoding. The embedder 114 uses the bitstream clock to clock in the data on the Data_in port.

The LN port is the line number during the vertical blanking interval of video where the closed caption data is to be encoded. In most applications, the line number will be 21.

The Add-parity port informs the embedder 114 to add an odd parity bit to the closed caption data before sending it out of the video decoder 44. The data for closed captioning or line 21 is 7 bits of data with one odd parity bit. It may not be desired to add the parity bit if data other than closed caption data or line 21 data is to be sent out of the video decoder 44 on the digital video outputs. Another instance where it is not desirable for the video decoder 44 to add the odd parity bit is when the analog video encoder 46 adds the odd parity bit itself when encoding the data.

The protocol select port informs the embedder 114 what type of protocol to use for placing the closed caption data on the digital video output ports. A very simple example of protocol may be to put only the two closed caption bytes out on the video output port on the first horizontal edge after a field transition. The closed caption data is always encoded on line 21 of the present field. This protocol may be adequate for the case when the video decoder 44 is directly connected to the analog video encoder 46.

An example of a flexible and more robust protocol involves generating a byte sequence that has a start sequence, line number, closed caption data and check sum values. The byte sequence may then be provided as an output at a certain time (determined by the protocol) during the vertical blanking interval. This exemplary protocol would be especially useful for embodiments in which the digital video out of the video decoder 44 passes through another module (such as a graphics overlay module) prior to the analog video encoder 46. The extra module between the video decoder 44 and the analog video encoder 46 at times may corrupt the data during the blanking interval. This may occur during graphics mode changes.

In a very robust and complicated protocol, the byte sequences are output with error correction more than once.

An example of the second protocol mentioned above will be described in more detail below. The embedder 114 produces a sequence of bytes that include closed caption data. Depending on the protocol established between the video decoder 44 and the analog video encoder 46, the sequence of bytes occurs at a certain time during the non-active video time. As an example, assume the sequence is: 0, 255, LN, CC_1, CC_2, Chk_sum1, Chk_sum2. The start sequence is represented by 0, 255. LN represents the line number where the closed caption data is to be encoded. CC_1 is the first byte of closed caption data. CC_2 is the second byte of closed caption data. Chk_sum1 and Chk_sum2 form the check sum of LN, CC_1 and CC_2. Chk_sum1 represents the 7 most significant bits of the check sum and Chk_sum2 the 7 least significant bits. The upper bit of Chk_sum1 and Chk_sum2 are each set to 0 to prevent false CCIR-656 start sequence emulation.

The start sequence is unique since, according to the EIA608 recommendation, all of the closed caption data types are ASCII. The data types are also limited to a range from 16 to 127.

The protocol may be a three-step process. The analog video encoder 46 makes a transition on the field signal. When the analog video encoder 46 generates the next falling edge of horizontal, it expects to receive the sequence of bytes shown above on its digital video input. If it does not see the above sequence of bytes with a valid check sum, then it assumes that there is no valid closed caption data available for encoding.

The Clear CC_data pin is brought low to clear the internal memory of the embedder 114. This is normally done whenever the user changes channels or program streams. The video clock is used to clock the eight-bit data out of the CC_data out port.

The CC_data out port is where the eight-bit data sequence that contains closed caption data exits the embedder 114. The closed caption data exits the port at the proper rate and at the proper time. In other words, there are only two closed caption bytes that are sent every odd field (and only two bytes sent for line 21 encoding per field). They occur at the right time, such as after the first horizontal edge after a field signal transition.

The Embed CC Select signal informs the second multiplexer 100 when the byte sequence containing closed caption data is exiting the embedder 114. When the data sequence is generated out of the embedder 114, the second multiplexer 130 sends the data sequence out on the digital video output port of the video decoder 44. At all other times, the Embed CC Select signal directs the second multiplexer 130 to send the digital video from the video decompressor out on the digital video output port of the video decoder 44.

The horizontal pin of the embedder 114 receives the horizontal synchronization signal. The field pin receives the field synchronization signal.

The second multiplexer 100 sends either the digital video from the video decompressor 102 or the byte sequence from the embedder 114 to the digital video output port of the video decoder 44. The selection is made through the select pin (the Embed CC Select signal from the embedder 114).

An embodiment of the embedder 114 to implement the exemplary second protocol is depicted in Figure 9. This embodiment includes a controller 122, a FIFO 124, an accumulator 128 and a multiplexer 130.

The controller 122 uses the FIFO 124 to store incoming closed caption data and meter this data out two bytes per field. The bitstream will not be running at the same clock rate as the Video_clock and the closed caption data into the embedder 114 my be burst. The bitstream clock is connected to the WR_CLK (write clock) pin of the FIFO 124, and CC_present is connected to the WR_EN (write enable pin of the FIFO 124). Data is written into the FIFO 124 on every rising edge of the bitstream clock whenever the CC_present signal is brought high. Data is read from the FIFO 124 on every rising edge of Video_clock whenever RD_EN (read enable) is high.

The accumulator 128 generates a check sum by adding LN and the two closed caption bytes. This number is more than eight bits so it is put in Chk_sum1 and Chk_sum2.

If the add_parity signal is high, then incoming closed caption data has an odd parity bit added to it before it is written into the FIFO 124. The controller 122 sets the Embedded CC Select signal high when it is sending out the embedded closed caption byte sequence.

The initial state of the controller 122 has the RD_EN and the ACC signals in their inactive low state. The clear_ACC signal is in its active low state. The

clear_FIFO signal is in its inactive high state. When it is supposed to send out the closed caption byte sequence, the controller 122 will perform the following sequence of actions.

On the first rising clock edge, a zero is sent out on the CC data out port by selecting the in0 port of the multiplexer 130. The clearing of the accumulator 128 is stopped by setting the clear_ACC signal high.

On the next rising clock edge, 255 is sent out on the CC data out port by selecting the in1 port of the multiplexer 130. The ACC signal is set high so that the accumulator 128 starts accumulating on the next rising clock edge. RD_EN is already low so that multiplexer 126 sends the LN byte to the accumulator 128.

On the next rising clock edge, the controller 122 sends LN out on CC data out port by causing the multiplexer 130 to select the in2 port. The RD_EN signal is set high. This allows the Video_clock read out the next closed caption data byte for multiplexers 126 and 130. Setting RD_EN high also causes multiplexer 126 to send the closed caption data from the FIFO 124 to the accumulator 128.

On the next rising clock edge, the controller 122 sends out the first closed caption byte from the FIFO 124 by causing the multiplexer 130 to select the in3 port. The reading of the closed caption data out of the FIFO 124 is stopped by setting RD_EN low. The accumulator 128 stops accumulating in response to the controller 122 setting the ACC signal low.

On the next rising clock edge, the first check sum byte (Chk_sum1) by causing the multiplexer 130 to select the in4 port.

On the next rising clock edge, the controller 122 sends out the second check sum byte (Chk_sum2) by causing the multiplexer 130 to select the in5 port. This is the end of the closed caption data sequence, so that on the

next rising clock edge, the controller 122 clears the accumulator 128 be setting the clear_ACC low. The controller 122 sends a 0 out on the CC data out port by selecting the in0 port of multiplexer 130.

When the clear_CC data is brought low, the controller 122 clears the contents of the FIFO 124 and the accumulator 128. The controller 122 returns to its initial state and waits for proper conditions (determined by the protocol select) before starting a new closed caption data byte sequence.

Another embodiment of a video decoder is depicted in Figure 10, this embodiment being particularly suitable in arrangements in which the video decoder 44' does not receive external horizontal and field signals for display synchronization. Instead, the video decoder 44' generates all synchronization signals. The video decoder of Figure 10 is used, for instance, with the analog video encoder 46 of Figure 5, and in the CCIR-656 mode, for example. The video decoder of Figure 10 is similar to that of Figure 8, so that description of elements common to these two embodiments will not be repeated.

The video decoder 44 in the embodiment of Figure 10 has a synchronization module 140 with a data encoding section that generates the SAV, EAV (start of active video, end of active video) and ancillary data. The synchronization module 140 controls multiplexer 100 to cause it to output one of: digital video from the video decompressor 102, the closed caption data byte sequences from the closed caption data embedder 114, or the CCIR-656 synchronization codes (SAV, EAV...).

The multiplexer 100 in this embodiment has three inputs: digital video from the video decompressor 102; CCIR-656 SAV, EAV information from the synchronization module 140; and the closed caption data byte sequence from the closed caption data embedder 114. The

synchronization module 140 informs the multiplexer 100 when to output the proper bytes.

In the embodiment of Figure 10, the closed caption data embedder 114 receives a signal 'CC_select' from the synchronization module 140 that informs the embedder 114 when to issue the closed caption byte sequence. Since it does not control multiplexer 100, the closed caption data embedder 114 in this embodiment does not generate the Embedded_CC_select output signal, in contrast to the embodiment of Figure 8. Additionally, the byte sequence out of the closed caption data embedder 114 in the embodiment of Figure 10 is 0, 254 instead of 0, 255. The values 0, 254 form the CCIR-656 start sequence for SAV and EAV from the synchronization module 140.

The embodiment of the closed caption data embedder 114 used in the embodiment of the video decoder of Figure 10 is depicted in Figure 11. Its operation is similar to the embodiment of the data embedder of Figure 9, with the exceptions as noted above in the description of the video decoder 44 of Figure 10.

Although the present invention has been described and illustrated in detail, it is clearly understood that the same is by way of illustration and example only and is not to be taken by way of limitation, the spirit and scope of the present invention being limited only by the terms of the appended claims.

CLAIMS

1. An arrangement for decoding an input transport signal, comprising:

a transport demultiplexer that receives an input transport stream and demultiplexes the input transport stream into an audio stream and a video stream containing compressed video and user data;

a video decoder that receives the video stream and generates decompressed video and closed caption data therefrom; and

an analog video encoder that receives the decompressed video and closed caption data and generates an analog video output signal with encoded closed caption data.

2. The arrangement of Claim 1, wherein the analog video encoder has a single interface for receiving the decompressed video and closed caption data.

3. The arrangement of Claim 2, wherein the closed caption data is encoded as ancillary data.

4. The arrangement of Claim 3, wherein the ancillary data includes code data which indicates that closed caption data follows the code data.

5. The arrangement of Claim 4, wherein the closed caption data follows the code data by a predetermined amount of bytes.

6. The arrangement of Claim 4, wherein the closed caption data follows the code data by an amount of bytes indicated by information in the code data.

7. The arrangement of Claim 4, wherein the analog video encoder includes a data decoder that examines the code data and prepares the analog video encoder to receive the closed caption data that follows the code data.

8. The arrangement of Claim 7, wherein the analog video encoder includes a closed captioning encoder and a multiplexer controlled by the data decoder to provide the closed caption data to the closed captioning encoder.

9. The arrangement of Claim 8, wherein the analog video encoder includes a chroma encoder that receives Cr and Cb data from the multiplexer and scales the Cr and Cb data for chroma encoding.

10. The arrangement of Claim 9, wherein the analog video encoder includes a luma processor that receives luma data from the multiplexer and interpolates the luma data.

11. The arrangement of Claim 3, wherein the decompressed video and closed caption data conforms to CCIR-656 standard.

12. The arrangement of Claim 2, wherein the analog video encoder includes a synchronization signal generator that generates at least one synchronization signal received by the video decoder, the video decoder sending the closed caption data to the analog video encoder in response to the at least one synchronization signal.

13. The arrangement of Claim 12, wherein the closed caption data is sent to the analog video encoder by the video decoder during a vertical synchronization period.

14. The arrangement of Claim 12, wherein the closed caption data is sent to the analog video encoder by the video decoder following a specific preamble of data bytes during a blanking interval.

15. The arrangement of Claim 12, wherein the analog video encoder includes a data decoder that examines data bytes to determine whether closed caption data is present.

16. The arrangement of Claim 15, wherein the analog video encoder includes a closed captioning encoder and a multiplexer controlled by the synchronization signal generator to provide to the closed captioning encoder the closed caption data determined by the data decoder to be present.

17. The arrangement of Claim 16, wherein the analog video encoder includes a chroma encoder that receives Cr and Cb data from the multiplexer and scales the Cr and Cb data for chroma encoding.

18. The arrangement of Claim 17, wherein the analog video encoder includes a luma processor that receives luma data from the multiplexer and interpolates the luma data.

19. The arrangement of Claim 18, wherein the decompressed video and closed caption data conforms to CCIR-601 standard.

20. An analog video encoder for converting a digital video stream to analog video signals, comprising:

a single interface for receiving a digital video stream that includes video data and closed caption data;

at least one video data processor for converting the video data into analog video signals;

a data decoder for identifying closed caption data in the digital video stream; and

a closed captioning encoder for receiving closed caption data and inserting the closed caption data into the analog video signals.

21. The analog video encoder of Claim 20, wherein the single interface includes a multiplexer that has an input for receiving the digital video stream, a plurality of selectable outputs, and a select input at which a select signal is received to select the output of the multiplexer.

22. The analog video encoder of Claim 21, wherein the at least one video processor includes a chroma encoder that receives Cr and Cb data in the video data from a first one of the outputs of the multiplexer, scales the Cr and Cb data for chroma encoding, and generates color burst information.

23. The analog video encoder of Claim 22, further comprising another video processor that is a luma processor that receives luma data from a second one of the outputs of the multiplexer and interpolates the luma data.

24. The analog video encoder of Claim 23, further comprising a digital to analog converter coupled to receive the outputs of the chroma encoder, the luma processor and the closed captioning encoder and converting the outputs to analog signals.

25. The analog video encoder of Claim 20, wherein the closed caption data received at the single interface is encoded as ancillary data.

26. The analog video encoder of Claim 25, wherein the ancillary data includes code data which indicates that closed caption data follows the code data.

27. The analog video encoder of Claim 26, wherein the closed caption data follows the code data by a predetermined amount of bytes.

28. The analog video encoder of Claim 26, wherein the closed caption data follows the code data by an amount of bytes indicated by information in the code data.

29. The analog video encoder of Claim 20, wherein the analog video encoder includes a synchronization signal generator that generates at least one synchronization signal to time sending of the closed caption data to the analog video encoder in the digital video stream.

30. The analog video encoder of Claim 29, wherein the synchronization signal is timed to cause receipt of the closed caption data by the analog video encoder during a vertical synchronization period.

31. The analog video encoder of Claim 29, wherein the synchronization signal is timed to cause receipt of the closed caption data by the analog video encoder following a specific preamble of data bytes during a blanking interval.

32. A method of inserting closed caption data in a video signal, comprising:

receiving digital video data and closed caption data at a single data interface of an analog video encoder; and

converting the digital video data and closed caption data into video analog signals with inserted closed captioning using the analog video encoder.

33. The method of Claim 32, wherein the step of receiving includes receiving the closed caption data encoded within ancillary data.

34. The method of Claim 33, wherein the ancillary data includes code data which indicates that closed caption data follows the code data.

35. The method of Claim 34, wherein the closed caption data follows the code data by a predetermined amount of bytes.

36. The method of Claim 35, wherein the closed caption data follows the code data by an amount of bytes indicated by information in the code data.

37. The method of Claim 32, further comprising generating at least one synchronization signal to time sending of the closed caption data to the analog video encoder in the digital video data.

38. The method of Claim 29, wherein the synchronization signal is timed to cause receipt of the closed caption data by the analog video encoder during a vertical synchronization period.

39. The method of Claim 29, wherein the synchronization signal is timed to cause receipt of the closed caption data by the analog video encoder following a specific preamble of data bytes during a blanking interval.

ANALOG VIDEO ENCODER WITH METERED CLOSED CAPTION DATA ON DIGITAL VIDEO INPUT INTERFACE

Abstract of the Disclosure

An arrangement and method for decoding an input transport signal, such as an MPEG transport signal, and inserting closed caption data presents an analog video encoder with decompressed video data and closed caption data at a single interface. This avoids polling by a microprocessor and the hardware and software expense.

Figure 1
Prior Art

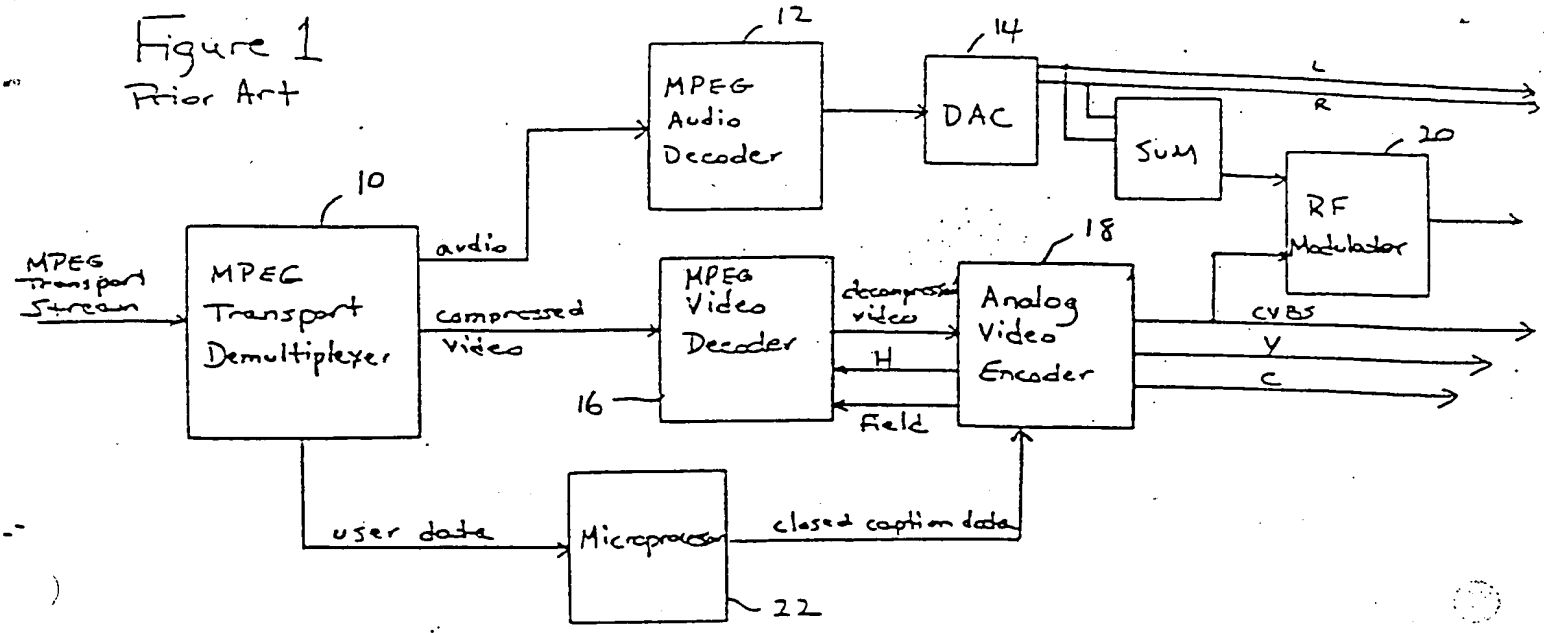
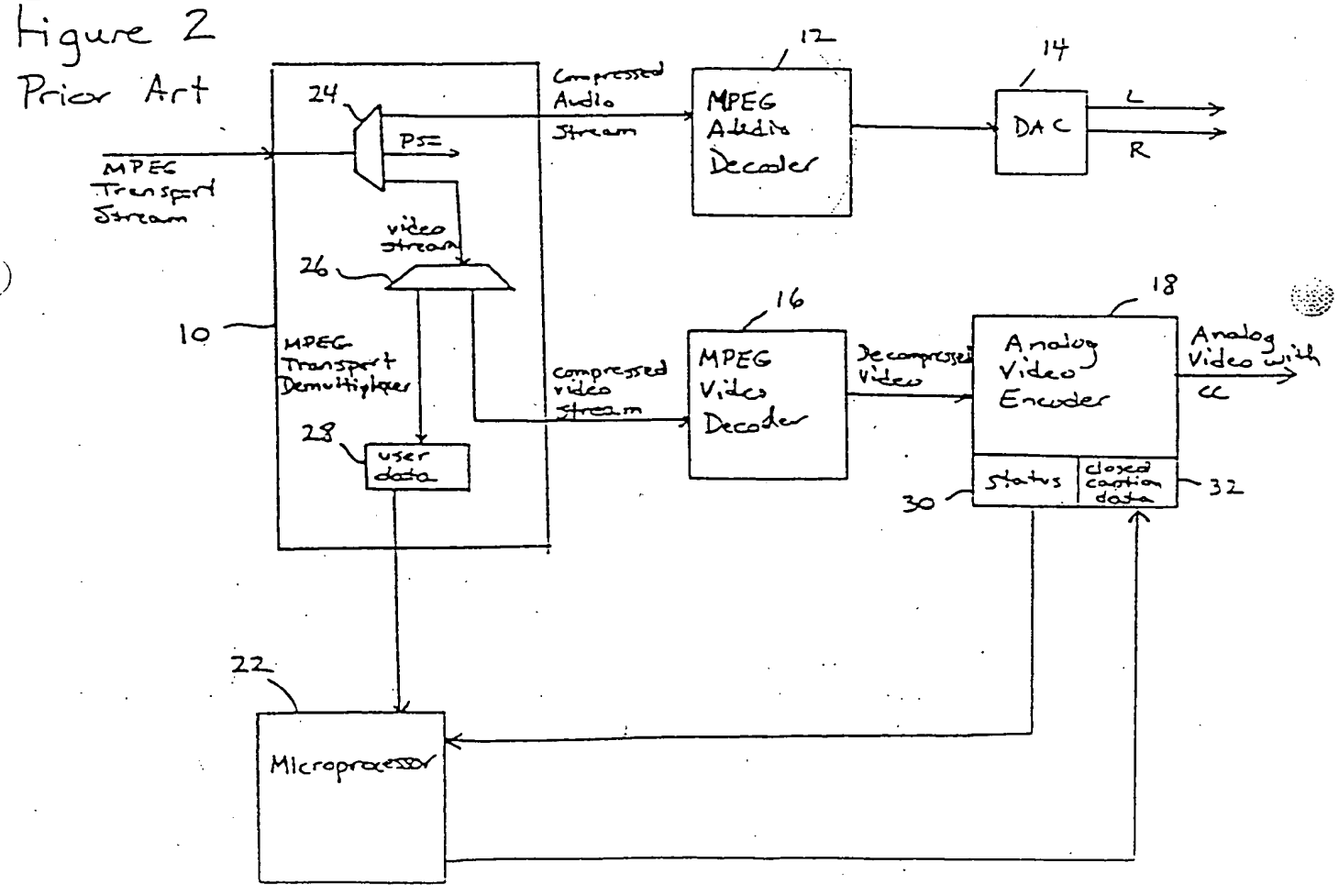


Figure 2
Prior Art



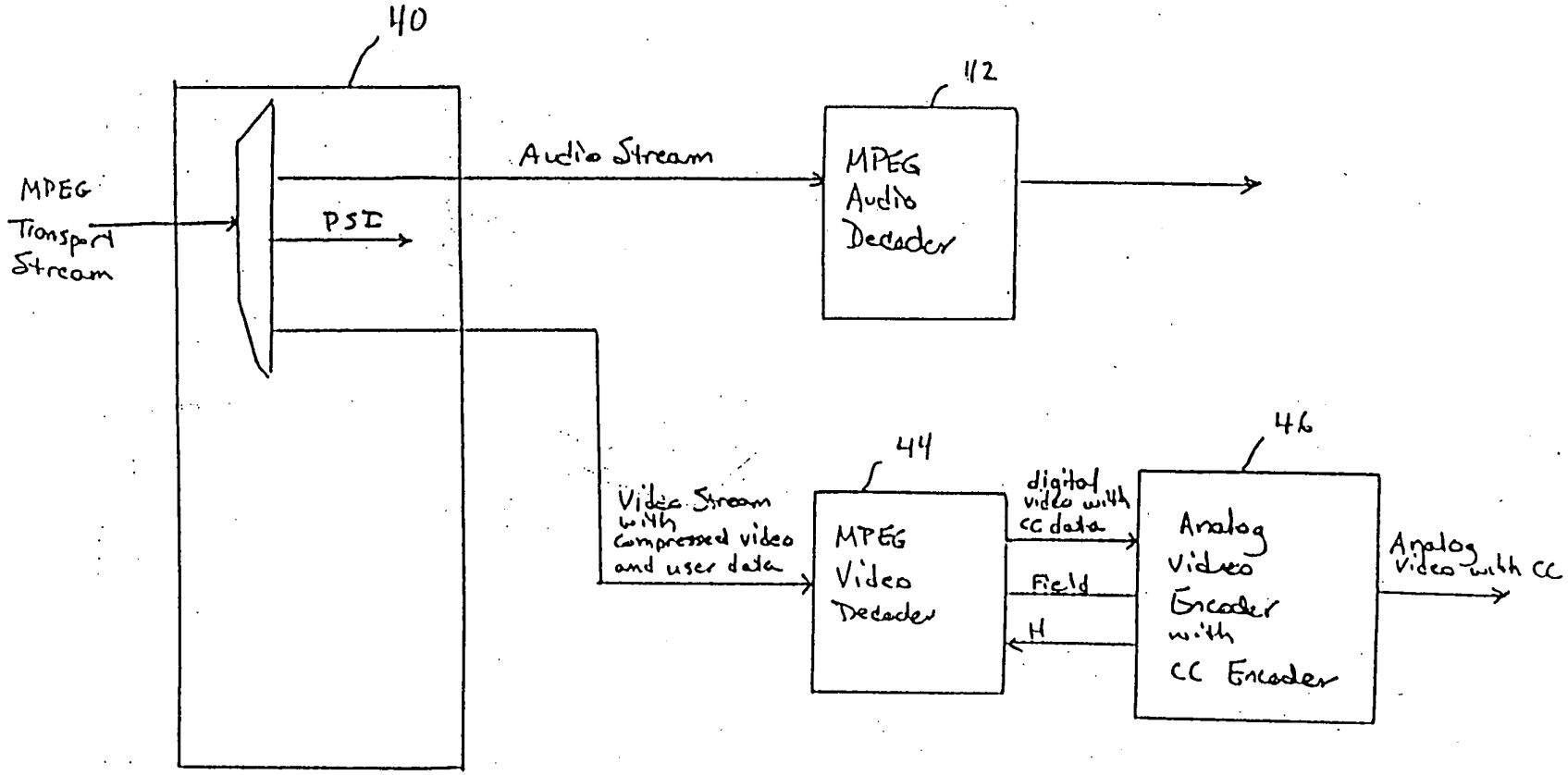

40
42
MPEG
Transport
Stream
PSI
Audio Stream
MPEG
Audio
Decoder
Video Stream
with
compressed video
and user data
44
MPEG
Video
Decoder
digital
video with
cc data
Field
H
46
Analog
Video
Encoder
with
CC Encoder
Analog
Video with CC

Figure 3

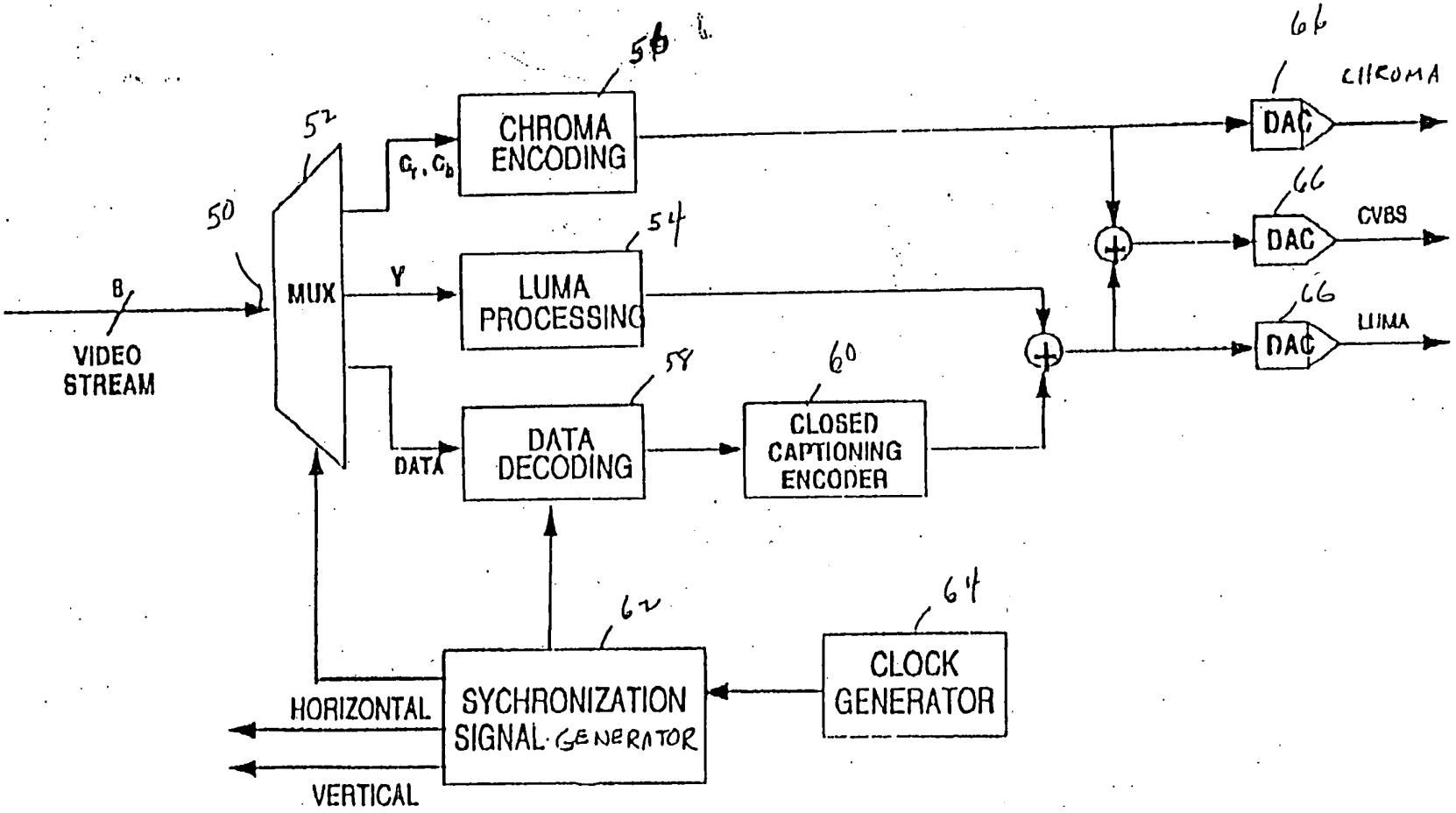

VIDEO STREAM
8
50
MUX
52
C_r, C_b
CHROMA ENCODING
56
Y
LUMA PROCESSING
54
DATA
DATA DECODING
58
CLOSED CAPTIONING ENCODER
60
SYCHRONIZATION SIGNAL GENERATOR
62
CLOCK GENERATOR
64
HORIZONTAL
VERTICAL
DAC
66
CHROMA
CVBS
LUMA

Figure 4

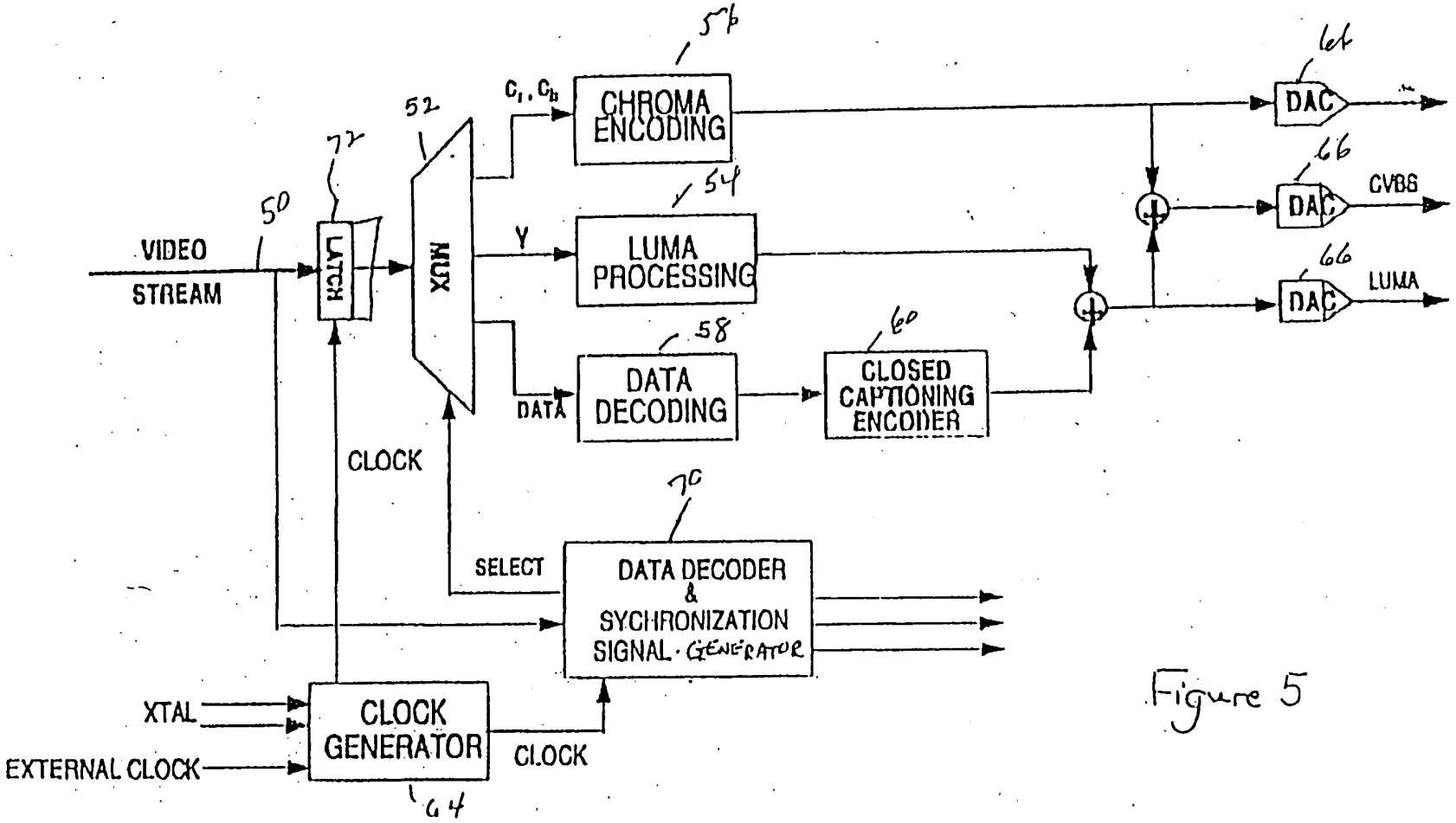

VIDEO
STREAM
50
LATCH
72
MUX
52
C_r, C_b
CHROMA ENCODING
56
Y
LUMA PROCESSING
54
DATA
DATA DECODING
58
CLOSED CAPTIONING ENCODER
60
DAC
64
DAC
66
CVBS
DAC
66
LUMA
CLOCK
SELECT
DATA DECODER & SYCHRONIZATION SIGNAL GENERATOR
70
XTAL
EXTERNAL CLOCK
CLOCK GENERATOR
64
CLOCK

Figure 5

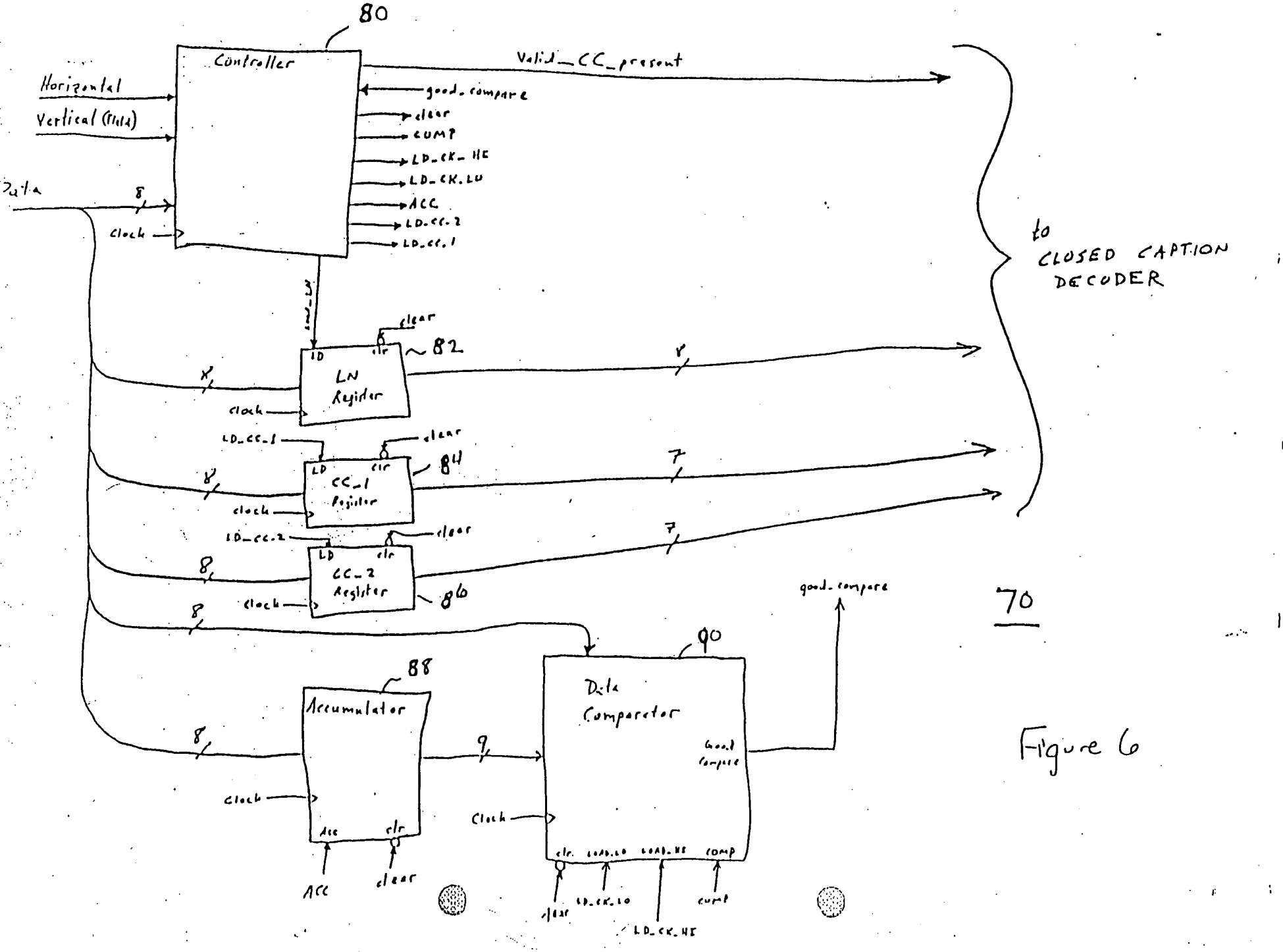

80
Controller
Horizontal
Vertical (Field)
Data
8
clock
Valid_CC_present
good_compare
clear
COMP
LD_CK_HI
LD_CK_LO
ACC
LD_cc_2
LD_cc_1
to CLOSED CAPTION DECODER
LD_cc_1
clear
LD
clr
LN Register
82
X
clock
Y
LD_cc_1
clear
CC_1 Register
84
clock
Y
7
LD_cc_2
clear
CC_2 Register
86
clock
8
7
8
good_compare
70
90
88
Accumulator
Data Comparator
8
9
clock
Acc
clr
ACC
clear
Clock
clr
LOAD_LO
LOAD_HI
COMP
clear
LD_CK_LO
LD_CK_HI
COMP

Figure 6

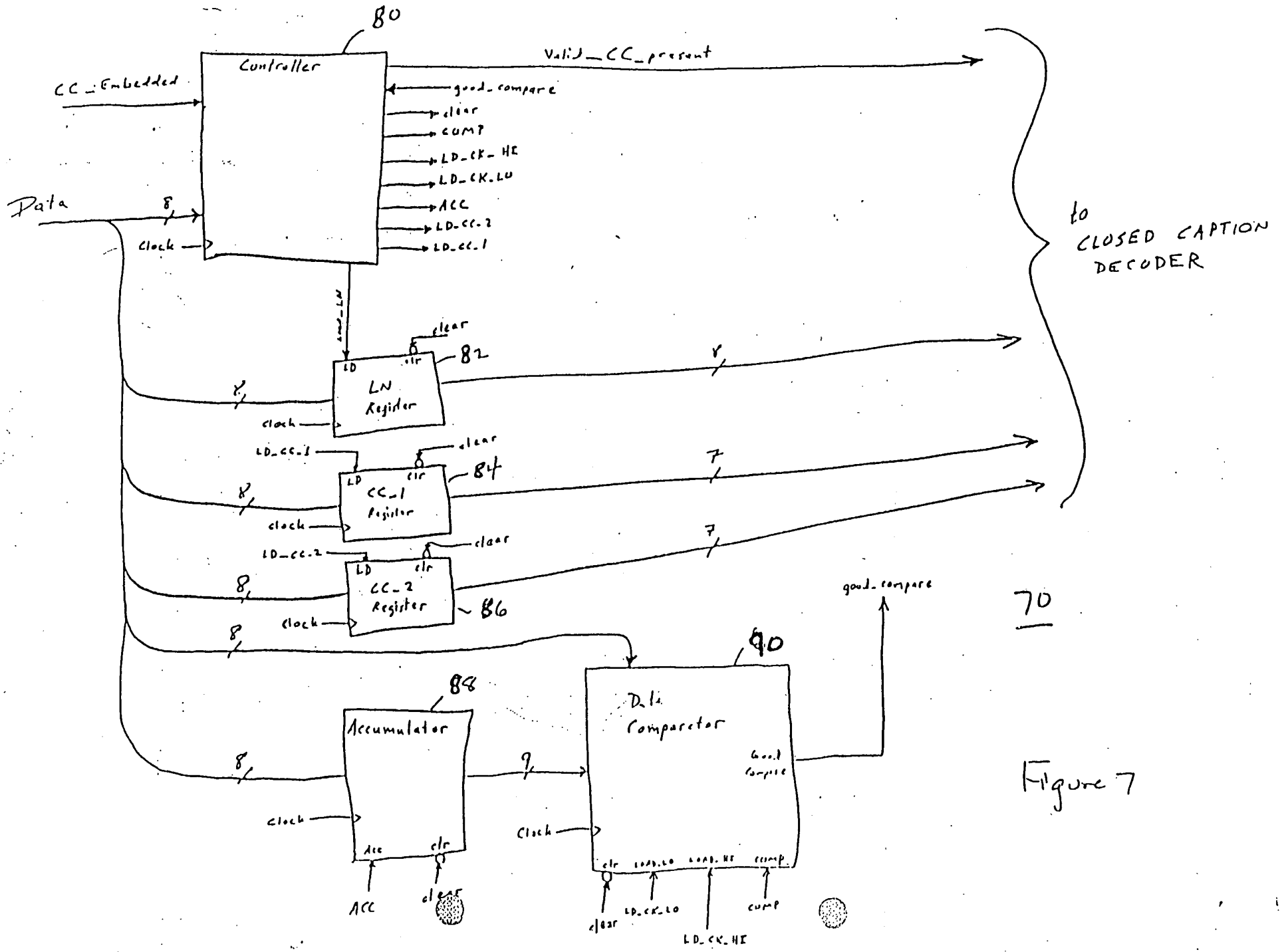

80
Controller
CC_Embedded
Data
8
Clock
Valid_CC_present
good_compare
clear
comp
LD_CK_HI
LD_CK_LO
ACC
LD_cc_2
LD_cc_1
to CLOSED CAPTION DECODER
LD_LN
clear
LD
clr
82
LN Register
8
clock
8
LD_cc_1
clear
84
CC_1 Register
7
LD_cc_2
clear
CC_2 Register
86
good_compare
70
90
88
Accumulator
9
Data Comparator
clock
clr
ACC
clear
LD_CK_LO
LD_CK_HI
comp

Figure 7

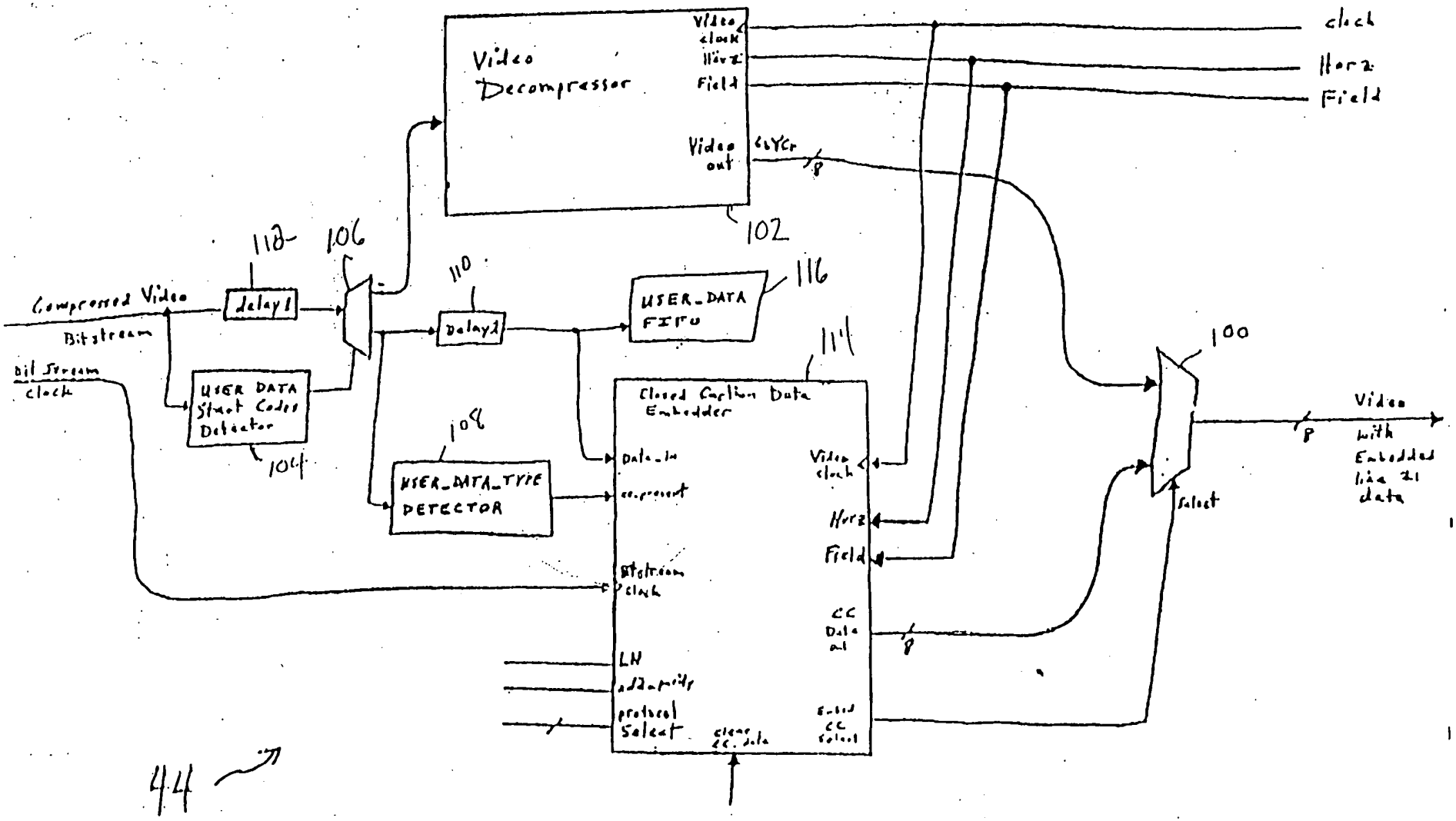

Video Decompressor
Video clock
Horz
Field
Video out
clock
Horz
Field
102
118
106
110
116
Compressed Video Bitstream
Bit Stream clock
Delay1
Delay2
USER_DATA FIFO
USER DATA Start Codes Detector
104
108
USER_DATA_TYPE DETECTOR
114
Closed Caption Data Embedder
Data_in
Video clock
Horz
Field
Bitstream clock
CC Data out
LN
protocol Select
clear CC data
Select CC select
100
Video with Embedded line 21 data
select
44

Figure 8

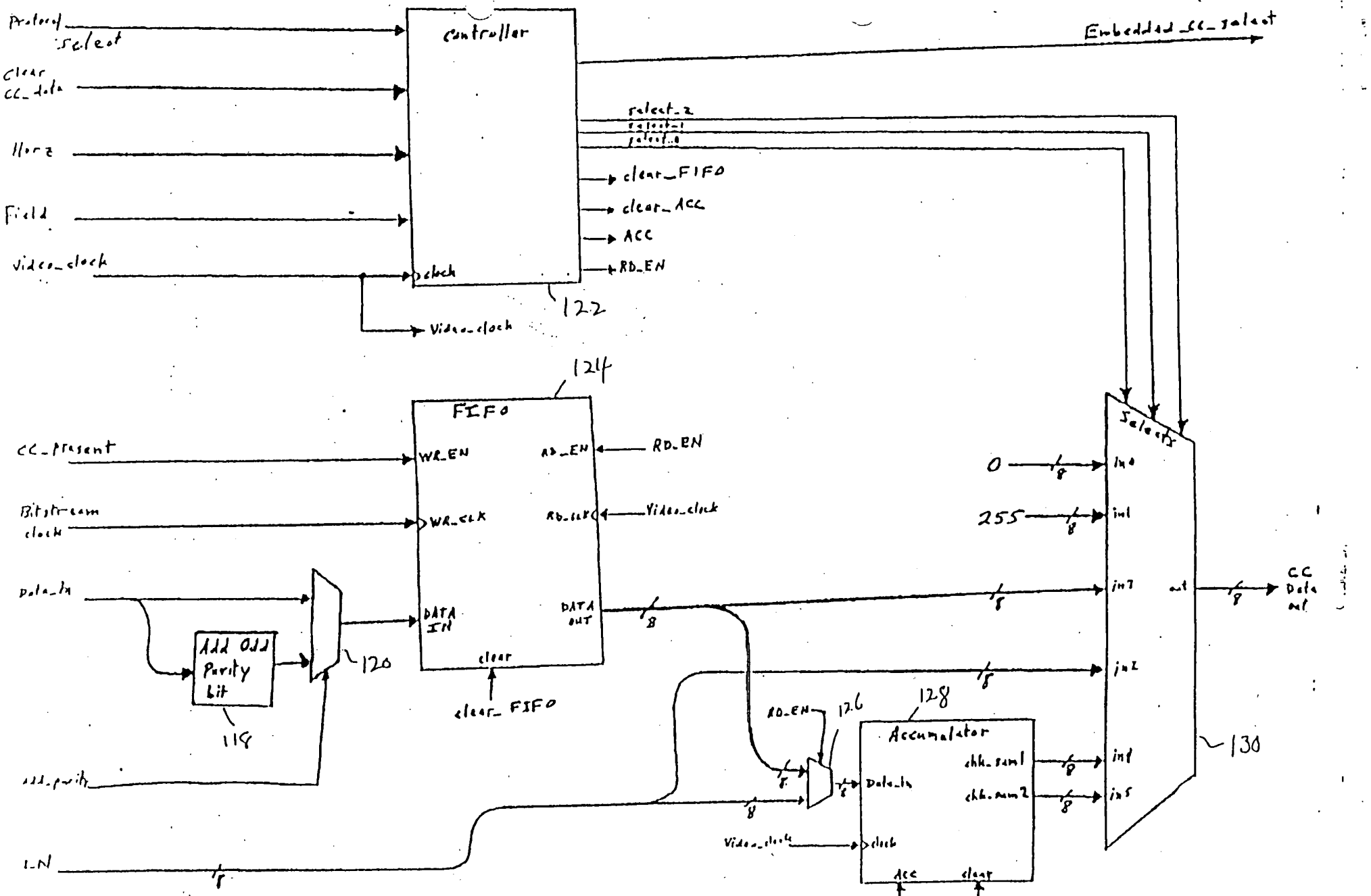

Protocol select
Controller
Embedded_CC_select
clear CC_data
Horz
Field
video_clock
clock
Video_clock
122
select_2
select_1
select_0
clear_FIFO
clear_Acc
ACC
RD_EN
124
FIFO
CC_present
WR_EN
RD_EN
Bitstream clock
WR_CLK
RD_CLK
Video_clock
Data_In
DATA IN
DATA OUT
clear
clear_FIFO
Add Odd Parity Bit
118
120
add_parity
I_N
RD_EN
126
128
Accumulator
Data_In
chk_sum1
chk_sum2
clock
Video_clock
Acc
clear
Acc
clear_Acc
Select
0
255
in0
in1
in7
in2
in1
in5
out
CC Data out
130

Figure 9

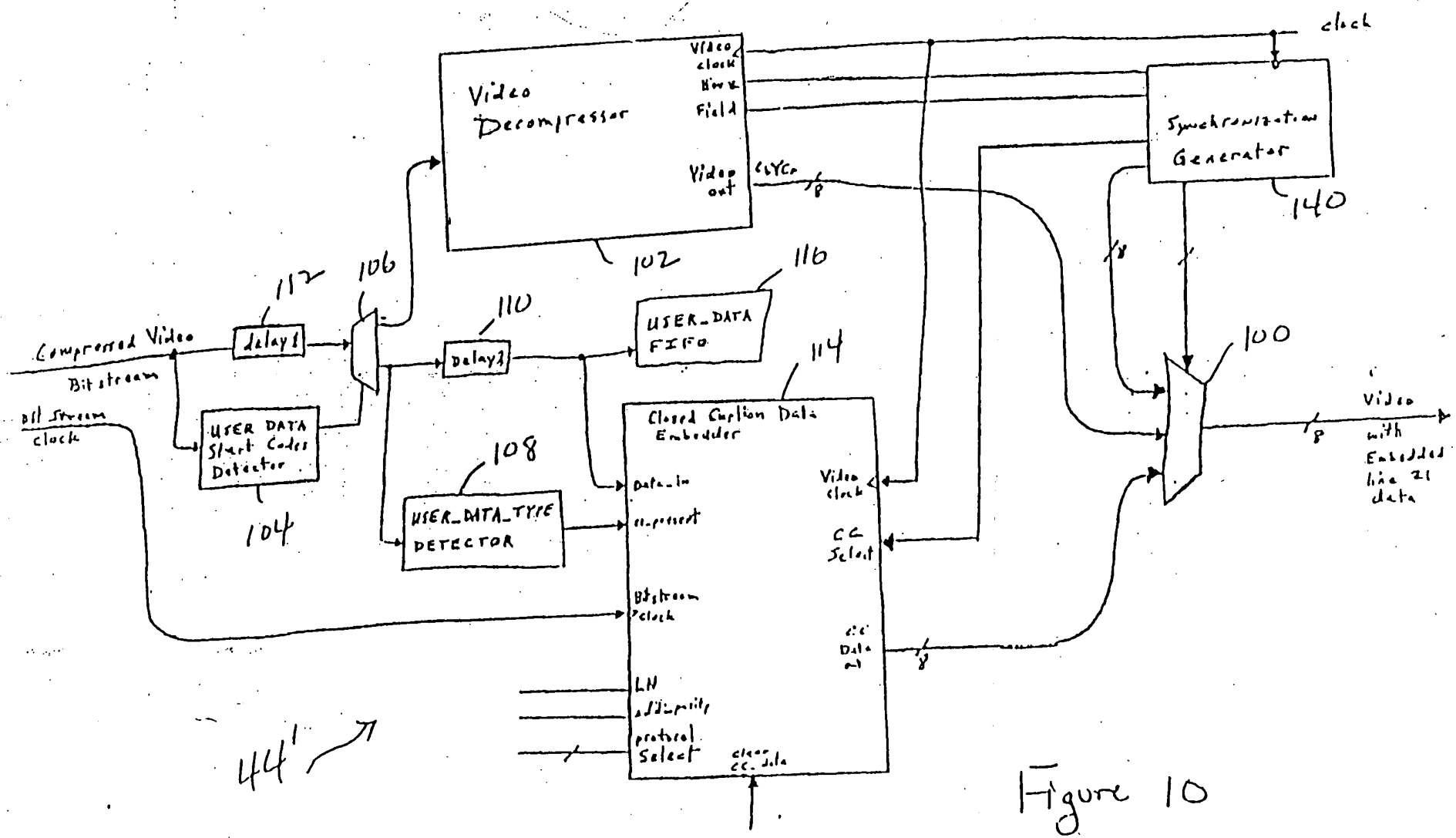
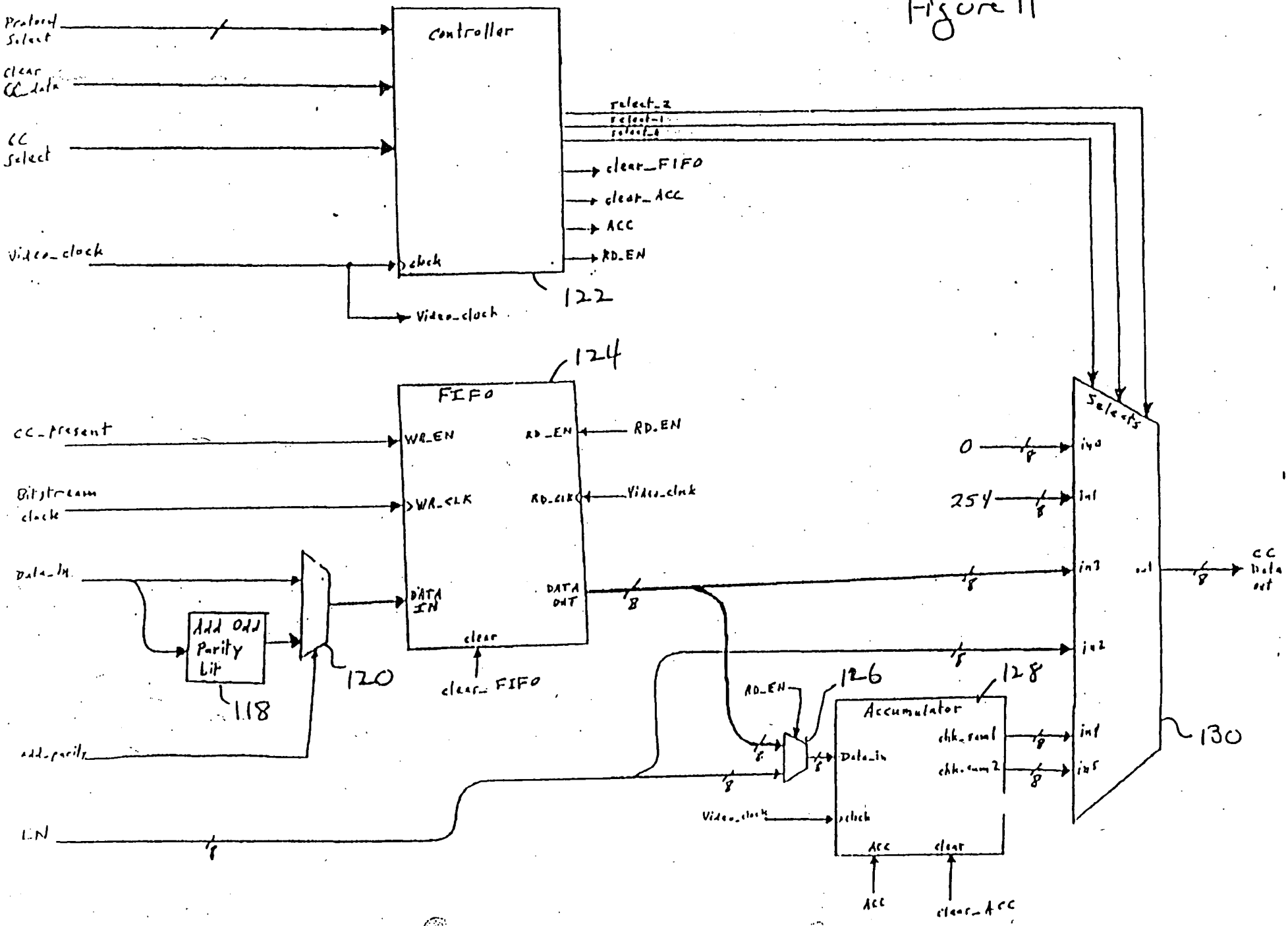

Video Decompressor
Video clock
Hsync
Field
Video out
CbYCr
102
clock
Synchronization Generator
140
116
USER_DATA FIFO
112
delay1
106
110
delay2
Compressed Video Bitstream
bit stream clock
USER DATA Start Code Detector
104
108
USER_DATA_TYPE DETECTOR
114
Closed Caption Data Embedder
data_in
u_present
Bitstream Pclock
LN
protocol Select
Video clock
CC Select
CC Data out
100
Video with Embedded line 21 data
44'

Figure 10

Figure 11


Protocol Select
clear CC_data
CC Select
Video_clock
Controller
clock
select_2
select_1
select_0
clear_FIFO
clear_ACC
ACC
RD_EN
Video_clock
122
124
FIFO
CC_present
Bitstream clock
Data_IN
WR_EN
WR_CLK
DATA IN
clear
RD_EN
RD_CLK
DATA OUT
Video_clock
clear_FIFO
Add Odd Parity Bit
118
120
add_parity
LEN
RD_EN
126
128
Accumulator
Data_in
clock
Acc
clear
chk_sum1
chk_sum2
Video_clock
ACC
clear_ACC
Selects
in0
in1
in2
in3
in5
0
254
out
CC Data out
130